VALTER CASTELHANO DE OLIVEIRA

PROPOSTA DE MÉTODO PARA GESTÃO DE REQUISITOS DE SISTEMAS INTEGRANDO MODELAGEM DE NEGÓCIO E LINGUAGENS FORMAIS

São Paulo
2008

VALTER CASTELHANO DE OLIVEIRA

PROPOSTA DE MÉTODO PARA GESTÃO DE REQUISITOS DE SISTEMAS INTEGRANDO MODELAGEM DE NEGÓCIO E LINGUAGENS FORMAIS

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para a obtenção do Título de Mestre em Engenharia.

São Paulo
2008

VALTER CASTELHANO DE OLIVEIRA

PROPOSTA DE MÉTODO PARA GESTÃO DE REQUISITOS DE SISTEMAS INTEGRANDO MODELAGEM DE NEGÓCIO E LINGUAGENS FORMAIS

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para a obtenção do Título de Mestre em Engenharia.

Área de Concentração: Engenharia Naval e Oceânica

Orientador: Prof. Livre-Docente José Reinaldo Silva

São Paulo
2008

Este exemplar foi revisado e alterado em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, de novembro de 2008

Assinatura do autor ____

Assinatura do orientador ____

FICHA CATALOGRÁFICA

Oliveira, Valter Castelhano de
Proposta de método para gestão de requisitos de sistemas integrando modelagem de negócio e linguagens formais / V.C. de Oliveira. -- ed.rev. -- São Paulo, 2008.
152 p.

Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Naval e Oceânica.

1.Engenharia de requisitos 2.Negócio (Modelagem) 3.Linguagens formais 4.UML I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Naval e Oceânica II.t.

DEDICATÓRIA

À minha esposa Eliana e aos meus filhos Letícia e Vitor, com amor e gratidão pela compreensão e apoio durante a elaboração deste trabalho.

AGRADECIMENTOS

Ao meu orientador, Prof. Dr. José Reinaldo Silva pela diretriz, confiança e inestimáveis sugestões que nortearam este trabalho.

À minha família, pelo apoio constante e compreensão nos momentos que tivemos de trocar nossa convivência pela dedicação a este trabalho.

A meus pais, pelo exemplo de vida, perseverança e dedicação aos estudos.

A minha cunhada Profa. Dra. Tereza Ribeiro de Freitas Rossi, pelo apoio, incentivo e paciência em nossas diversas discussões;

Ao Departamento de Engenharia Naval e Oceânica da escola Politécnica da USP pela oportunidade de realização deste trabalho.

Aos funcionários do Departamento de Engenharia Naval e Oceânica da escola Politécnica da USP pela disponibilidade e constante atenção.

RESUMO

Apesar das novas e efetivas técnicas de engenharia de software, os projetos de desenvolvimento de sistemas estão propensos a ter os mesmos problemas que acometem o software de apoio à gestão. Entrega com atraso, acima do orçamento e não suprindo as reais necessidades dos usuários finais ou da organização que está financiando o desenvolvimento do sistema, são os principais problemas. Esse último problema é o que mais afeta o desenvolvimento de sistemas e é um desafio para que o desenvolvimento personalizado seja uma solução real para várias empresas. Este trabalho apresenta uma proposta de método de gestão que auxilie a comunicação entre as atividades associadas à engenharia de requisitos e as atividades associadas à modelagem dos processos de negócio. Essa abordagem concerne à gestão e tratamento de requisitos de sistemas baseando-se em técnicas de engenharia de processos de negócios e de engenharia de requisitos, no processo unificado de desenvolvimento de software e na utilização de linguagens semi-formais e formais de modelagem, UML e SysML respectivamente. O método pretende mitigar os efeitos dos problemas de comunicação existentes entre os diversos integrantes de um projeto, com especial atenção para a comunicação entre a equipe de requisitos do projeto e os *stakeholders* responsáveis pela aceitação e aprovação do sistema. A pesquisa, com o apoio da apresentação de dois casos que ilustram o método de gestão proposto, permite concluir que é possível tornar mais efetiva e produtiva a comunicação entre os diversos envolvidos com o projeto, podendo resultar em um processo mais eficiente para a aceitação dos requisitos junto aos *stakeholders*.

Palavras-chave: Engenharia de Requisitos. Modelagem de Negócio. Linguagens Formais. UML. SysML.

ABSTRACT

Despite new and effective software engineering techniques, system development projects are likely to have the same problems that affect the management support software. Delivery delay, above budget and not fitting the real needs of end users or the organization that is funding the system development, are the most common problems. The latter problem is the one that most affects the systems development and is a challenge for the custom development to be a real solution to several companies. This work presents a proposal for a management method to help the communication between the activities associated with the engineering requirements and the activities associated with business processes modeling. This approach, concerns to the systems requirements treatment and management, is based on business processes engineering and requirements engineering, in software development unified process and in the use of semi-formal and formal modeling languages as UML and SysML, respectively. The method seeks to mitigate the effects of the communication problems among the project members, with special attention to the communication between the project requirements team and the stakeholders responsible for the system acceptance and adoption. The research, supported by the presentation of two cases which illustrates the proposed management method, has concluded that it is possible to make more effective and productive communication among members related with the project, which may result a more efficient process for the stakeholders requirement acceptance.

Keywords: Requirements Engineering. Business Modeling. Formal Languages. UML. SysML.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

BPMS	Business Process Management Systems
BPP	Business Process Patterns
CASE	Computer-aided software engineering
CIM	Computer Integrated Manufacturing
CIMOSA	Computer Integrated Manufacturing Open System Architecture
CORE	Controlled Requirement Expression
ESPITI	European Software Process Improvement Training Iniciative
IDEF	Integrated Definition
INCOSE	International Council on Engineering Systems
JAD	Joint Application Design
MDA	Model Driven Architecture
MDD	Model Driven Development
MOF	Meta-Object Facility
OMG	Object Management Group
OSTN	Object State Transition Network
RAD	Rapid Application Development
RSA	IBM Rational Software Architect
RSD	IBM Rational Systems Developer
RSM	IBM Rational Software Modeler
RFP	Request for Proposal
RUP	Rational Unified Process
SADT	Structured Analysis and design Technique
SysML	Systems Modeling Language
UML	Unified Modeling Language

VM	Vending Machine
VOSE	Viewpoint-oriented System Engineering
VORD	Viewpoint-oriented System Definition
XMI	XML Metadata Interchange
XML	Extensible Markup Language

SUMÁRIO

1 INTRODUÇÃO

Segundo Booch et al (1999), uma organização bem sucedida no desenvolvimento de sistemas é aquela que consegue obter produtos de qualidade que casam com as necessidades de seus usuários de forma adequada e previsível, através da utilização efetiva e eficiente dos seus recursos humanos e materiais.

O desenvolvimento de sistemas requer a participação de pessoas com os mais diversos perfis e expectativas quanto aos resultados obtidos, oferecendo instrumentos eficientes de comunicação entre estas pessoas visando ao entendimento adequado de suas expectativas. O usuário final espera um sistema que atenda as suas necessidades diárias e facilite o seu trabalho; os financiadores do sistema esperam um sistema que ofereça o retorno adequado a seu investimento; e a equipe de desenvolvimento espera desenvolver um sistema de qualidade, no prazo e custo estipulado e atendendo as reais necessidades dos usuários (Leffingwell & Widrig, 2000).

Apesar das novas e efetivas técnicas de engenharia, os projetos de desenvolvimento de sistemas estão propensos a falhar e freqüentemente sistemas complexos são entregues com atraso, acima do orçamento e não atendem as reais necessidades dos usuários finais ou da organização que está financiando o sistema (Kotonya & Sommerville, 1998).

Segundo Procaccino et al (2002), estudos realizados em empresas americanas indicaram que cerca de 20% dos projetos de desenvolvimento de sistemas fracassaram e ainda segundo Molokken & Jorgensen (2003) a maioria dos projetos de software, entre 60 e 80%, foram executados com atraso ou orçamento excedido.

No processo de desenvolvimento de sistemas é geralmente das etapas iniciais que decorrem a maior parte dos problemas. Paradoxalmente as etapas iniciais são as menos dispendiosas por não exigirem grande aquisição de bens e serviços. Porém, são as de maior impacto negativo quando realizadas de forma inadequada. Estas etapas constituem o processo de engenharia de requisitos, que tem sido reconhecida como uma das mais importantes fases do processo de engenharia de sistemas (Kotonya & Sommerville, 1998).

Os processos associados à engenharia de requisitos são os responsáveis pelas evidências mais comuns e sérias de problemas, sendo 12% destes problemas associados a mudanças nos requisitos e especificações, 12% associados a requisitos e especificações incompletos e 13% associados à falta de engajamento do usuário (Leffingwell & Widrig, 2000).

O custo para reparar um problema no desenvolvimento de sistemas aumenta significativamente quando este é descoberto nas etapas mais avançadas do ciclo de vida do desenvolvimento. O custo para tratar um problema detectado quando o sistema está em produção é 500 vezes maior do que se fosse detectado e tratado na fase de requisitos (Carr, 2000).

De acordo com Kotonya & Sommerville (1998), a atividade final da engenharia de requisitos é a validação de requisitos que objetiva estabelecê-los de forma correta, consistente, integra e precisa. A validação de requisitos é um processo longo, envolvendo um grupo heterogêneo de pessoas que estão tratando um extenso e complexo documento, envolvendo a participação dos *stakeholders*[1], dos engenheiros de requisitos e da equipe de *design*, através da análise dos requisitos

---

[1] *Stakeholders* são as pessoas ou organizações que serão afetadas pelo sistema e que influenciam de forma direta ou indireta os requisitos do sistema (Kotonya & Sommerville, 1998).

documentados, buscando problemas, omissões e ambigüidades. O principal problema na validação dos requisitos é a inexistência de uma referência documentada para ser utilizada como base na validação do documento de requisitos. (Kotonya & Sommerville, 1998).

A utilização dos processos de negócio como referência documentada pode oferecer uma base de aderência para a validação e aceitação dos requisitos do sistema, fornecendo elementos para a busca de um método de gestão de requisitos capaz de catalisar, de forma eficiente, a complexa e exaustiva atividade de validação e aceitação dos requisitos junto aos usuários e *stakeholders* envolvidos com o sistema. A adoção de uma linguagem de modelagem formal, precisa e padronizada pode oferecer a este método de gestão de requisitos a eliminação de possíveis ambigüidades e imprecisões nos modelos, podendo ser utilizada pelas diversas equipes do projeto, oferecendo um canal básico e consistente para a comunicação interna e externa do projeto.

1.1 OBJETIVOS

O objetivo deste trabalho é propor um método de gestão que auxilie a comunicação, durante as etapas de definição dos requisitos de um sistema, entre a equipe de engenharia de requisitos e a equipe de engenharia de processos de negócio, através de uma maior aderência entre os requisitos do sistema e o modelo de negócio da organização.

Especificamente o trabalho se propõe a:

- Apresentar os fundamentos teóricos da Engenharia de Processos de Negócio e da Engenharia de Requisitos, de modo a identificar os pontos críticos

relevantes para a aderência entre a modelagem dos processos de negócio e a eliciação e especificação dos requisitos.

- Propor um método de gestão de requisitos sustentado pela aderência com os processos de negócio da empresa, que na fase de eliciação e aceitação dos requisitos do sistema, auxilie a comunicação com os *stakeholders*.
- Buscar uma linguagem formal de especificação de requisitos que permita a gestão dos requisitos aderente aos processos de negócio.

1.2 JUSTIFICATIVA

Considerando-se que:

- A engenharia de sistemas possui um enfoque interdisciplinar que viabiliza a realização com êxito de sistemas através da definição antecipada das necessidades e funcionalidades requeridas, da documentação dos requisitos, da síntese do *design*, da validação do sistema e da consideração do problema como um todo (INCOSE, 2007).
- A modelagem de sistemas é um elemento importante do processo de engenharia de sistemas, pois oferece, aos diversos envolvidos no negócio, uma abstração capaz de facilitar o entendimento de processos complexos de negócio, oferecendo instrumentos que viabilizem a busca e obtenção de melhorias e inovações (Noran, 2000).
- A complexidade dos processos de negócio das organizações implica a utilização de estruturas e modelos adequados para a gestão dos processos de negócio e que as necessidades impostas por estes processos de negócio devem direcionar as especificações dos sistemas nas empresas (Marshall, 2000).

- A melhoria da comunicação entre os diversos integrantes de um projeto, principalmente entre as equipes de especificação de requisitos, modelagem de negócio e os *stakeholders* responsáveis pela aceitação e aprovação dos requisitos do sistema, pode resultar em melhoras acentuadas na qualidade dos artefatos gerados nas etapas de especificação de requisitos.

Tem-se que a proposta do método de gestão de requisitos, aderente a modelagem de negócio da empresa e utilizando linguagem formal de modelagem e especificação, além de ser inovadora dentro da literatura científica da engenharia de requisitos e da engenharia de processos de negócio, pode alterar a forma como as equipes de trabalho abordam e registram, tanto a especificação dos requisitos como a modelagem de negócio, oferecendo uma possível redução de custo e tempo, principalmente nas fases de validação e aceitação dos requisitos, e uma possível melhora na aderência entre o sistema final e as reais necessidades das empresas e dos *stakeholders*, aumentando assim as expectativas de sucesso nos projetos de desenvolvimento de sistemas.

1.3 ORGANIZAÇÃO DO TRABALHO

Considerando o objetivo deste trabalho e as principais questões a serem estudas para atender este objetivo, foi proposta uma pesquisa aplicada, pois objetiva gerar conhecimentos para aplicação prática e dirigidos à solução de problemas específicos, envolvendo verdades e interesses locais. Além disso, este estudo busca uma melhor compreensão do foco abordado pela motivação, podendo ser classificada como sendo exploratória com relação aos seus objetivos, pois visa a proporcionar maior familiaridade com o problema, visando a torná-lo explícito e finalmente, pesquisa qualitativa, pois considera que nem tudo pode ser quantificado

e traduzido em números, onde a interpretação dos fenômenos e a atribuição de significados são elementos básicos e que existe uma relação subjetiva e dinâmica entre o mundo real e o objeto de estudo.

No capítulo 2 é feita uma revisão da literatura, apresentando conceitos das diversas abordagens voltadas à modelagem de processos do negócio, à engenharia de requisitos, ao desenvolvimento de sistemas de informação e à utilização de linguagens formais e semi-formais destinadas à especificação de sistemas.

O capítulo 3 apresenta à proposta de um método para gestão de requisitos, aderente ao modelo de processos de negócio e utilizando linguagens formais para a representação dos requisitos.

O capítulo 4 é dedicado a uma ilustração do método proposto através de dois estudos de caso: o primeiro associado ao desenvolvimento de sistemas de informação e o segundo associado ao desenvolvimento de um sistema de supervisão e controle de máquinas de venda.

No capítulo 5 são discutidos os pontos relevantes sobre o método de gestão proposto e sua aplicação prática aos casos.

E, finalmente, as conclusões a respeito do estudo desenvolvido e possíveis encaminhamentos para trabalhos futuros são tratados no capítulo 6.

2 REVISÃO DA LITERATURA

Considerando o objetivo deste trabalho que é desenvolver um método de gestão que auxilie a comunicação entre as atividades associadas à engenharia de requisitos e as atividades associadas ao levantamento de processos de negócio, o foco deste capítulo é apresentar um panorama sobre a pesquisa desenvolvida, mais especificamente através da abordagem dos seguintes itens:

- Apresentar conceitos básicos associados à engenharia de sistemas, originária da engenharia de processos de negócio e da engenharia de requisitos.
- Abordar a engenharia de processos de negócio buscando destacar a importância destes processos para as organizações, a diversidade de técnicas e métodos de tratamento e representação, com ênfase na utilização de UML para modelagem de negócio.
- Evidenciar a importância do processo unificado de desenvolvimento de software, que contempla desde as atividades de modelagem de negócio até as atividades de teste e implantação.
- Introduzir o conceito de engenharia de requisitos e sua importância dentro do processo de desenvolvimento de sistemas, descrever as etapas do processo de engenharia de requisitos e destacar a importância da eliciação, validação e gestão dos requisitos.
- Introduzir os conceitos associados à linguagem formal SysML, estendida da UML 2 e criada para ser uma linguagem unificada de modelagem de sistemas.

2.1 ENGENHARIA DE SISTEMAS

A engenharia de sistemas focaliza os diversos elementos que compõem um negócio, analisando-os, projetando-os e organizando-os em um sistema que pode ser um produto, um serviço ou uma tecnologia para a transformação da informação (Pressman, 2006).

De acordo com o INCOSE, *International Council on Engineering Systems*, a engenharia de sistemas pode ser definida como:

> "[...] enfoque interdisciplinar que viabiliza a realização com êxito de sistemas. Possui foco na definição antecipada, dentro do ciclo de desenvolvimento, das necessidades dos clientes e das funcionalidades requeridas, documentando os requisitos, para então passar a fase de síntese do *design* e validação do sistema, considerando o problema como um todo. A Engenharia de Sistemas integra todas as disciplinas e grupos especializados em uma única equipe, criando um processo estruturado para o desenvolvimento que passa da concepção, para a produção e depois para a operação. A Engenharia de Sistemas considera tanto as necessidades de negócio, como as necessidades técnicas de todos os clientes com o objetivo de fornecer um produto de qualidade atendendo as necessidades do usuário." (INCOSE, 2007)

A engenharia de sistemas toma diferentes formas dependendo do domínio da aplicação. Quando o contexto do trabalho de engenharia focaliza uma empresa de negócios, tem-se a engenharia de processo de negócio. Quando o contexto é a construção de um produto, tem-se o processo de engenharia de produto. Em ambos os contexto no qual o sistema reside, este deve ser entendido e uma representação efetiva produzida (Pressman, 2006).

A modelagem de sistemas é um elemento importante do processo de engenharia de sistemas, independente da visão considerada, pois oferece, aos diversos envolvidos no negócio, uma abstração capaz de facilitar o entendimento de processos complexos de negócio, oferecendo instrumentos que viabilizem a busca e obtenção de melhorias e inovações (Noran, 2000).

Um modelo é uma simplificação da realidade, construído para melhorar o entendimento do sistema ou negócio. O processo de modelagem permite o refinamento sucessivo da representação de um problema, obtido através da decomposição em sub-sistemas, oferecendo recursos para a representação de sistemas complexos, através da aplicação de alguns princípios básicos (Booch et al, 1999) (Leffingwell & Widrig, 2000):

- A escolha de qual modelo criar tem uma profunda influência do modo como o problema é atacado e de como a solução é formatada.
- Cada modelo pode ser expresso em diferentes níveis de precisão e detalhamento.
- Os melhores modelos estão conectados com a realidade.
- Nenhum modelo é suficiente. Cada sistema um pouco mais complicado é mais eficientemente representado por um pequeno conjunto de modelos independentes.
- Distribuição e partição de funcionalidades são otimizadas para atingir totalmente as funcionalidades do sistema com mínimo custo e máxima funcionalidade.

O engenheiro de sistemas modifica a influência relativa de diferentes elementos do sistema (pessoal, hardware e software) para derivar os diversos tipos de modelos, sempre considerando vários fatores restritivos como premissas, simplificações, limitações, restrições e preferências (Pressman, 2006).

A engenharia de sistemas está dividida em duas partes: caracterização do problema e análise do sistema. As atividades de caracterização do sistema tratam da criação, implementação e modificação dos sistemas. As atividades associadas à análise do sistema tratam da escolha, entre várias alternativas possíveis, de solução

do problema. O modelo obtido no processo de engenharia de sistemas pode adotar uma solução completamente automatizada, uma solução semi-automatizada ou um enfoque não automatizado (Kintschner, 2003) (Pressman, 2006).

A engenharia de sistemas oferece um conjunto de princípios básicos a serem utilizados como técnica de análise. São eles (INCOSE, 07) (Leffingwell & Widrig, 00):

- Conhecer o problema, conhecer o cliente e conhecer o usuário.
- Utilizar critérios baseados em necessidades efetivas para as decisões do sistema.
- Estabelecer e gerenciar os requisitos;
- Identificar e avaliar alternativas que sejam convergentes para a solução.
- Verificar e validar requisitos e soluções de desempenho.
- Manter a integridade do sistema.
- Utilizar um processo de documentação adequado.
- Gerenciar de acordo com o plano.

2.2 ENGENHARIA DE PROCESSOS DE NEGÓCIO

Desde os primórdios da humanidade, o homem busca produzir bens e serviços que promovam melhoria em sua qualidade de vida. Com a introdução do dinheiro em substituição aos mecanismos de troca de mercadorias, surgiram novos e sofisticados métodos para tratamento das mercadorias, seus custos e preços, e respectiva contabilização, incrementando, de forma significativa, a complexidade dos processos de negócio das organizações. (Marshall, 2000).

A intensa concorrência e as pressões econômicas sobre as grandes organizações no final do século passado mostraram que a iniciativa de melhoria de qualidade e de melhoria contínua e paulatina dos processos, embora essenciais,

não são suficientes para a manutenção dos negócios. O desempenho comercial está condicionado a uma abordagem revolucionária que deve abranger tanto a maneira de ver e estruturar a atividade, como também a maneira de melhorá-la. As atividades empresariais precisam ser tratadas como processos-chave ao invés de funções, departamentos e produtos. A realização de níveis de melhoria dessa ordem de magnitude em tais processos significa seu replanejamento do princípio ao fim, com emprego de todas as tecnologias inovadoras e recursos organizacionais disponíveis (Davenport, 1994).

A complexidade dos processos de negócio das organizações, que envolvem um grande número de entidades e pessoas, implica a utilização de estruturas e modelos adequados para a gestão destes processos, mantendo o foco na finalidade da organização, mantendo assim, a organização viva e lucrativa. Organizações de sucesso desenvolvem estratégias para definir e atingir suas finalidades. A finalidade de uma organização abrange sua visão, missão, metas e objetivos, os quais são alcançados através dos processos de negócio. Os processos de negócio devem abranger também os relacionamentos e comunicações entre diferentes organizações, permitindo que trabalhem juntas e se comuniquem (Rukanova et al, 2005) (Marshall, 2000).

Um processo de negócio requer entidades, tais como, materiais, pessoas, equipamentos e tecnologia, os quais a organização deve adquirir, desenvolver e organizar. Estas entidades estão inter-relacionadas, e a organização necessita gerenciá-las em seus processos de negócio com o objetivo de atingir a finalidade principal da organização (Marshall, 2000).

2.2.1 Processos de negócio

Definições sobre processo são encontradas em diversos ramos da ciência e sempre com significados semelhantes, embora sejam tratados em assuntos diferentes, como segue:

- Processo representa o fluxo de trabalho e informações existente durante todo o negócio. Estes processos atuam nas entidades de negócio para realizar as funções de negócio. Processos de negócio podem ter longa duração ou curta duração. Processos de negócio de reengenharia tipicamente são de longa duração (OMG, 2007).
- Processo é uma ordenação específica das atividades no tempo e espaço, com começo e fim identificado, entradas e saídas claramente identificadas, ou seja, uma estrutura para a ação (Davenport, 1994).
- Processos de negócio são sistemas sociais e tecnológicos, executados por pessoas e máquinas, e apoiados por sistemas de gerenciamento de processos de negócio (Shaw et al, 2007).
- Processo de negócio define como uma organização atinge suas finalidades. Processos estratégicos desenvolvem a visão e a missão, processos táticos e operacionais são utilizados para atingir as metas e os objetivos da organização. Um processo de negócio pode abranger um simples processo linear com poucos passos ou uma rede complexa de atividades interdependentes (Marshall, 2000).
- Os processos de negócio oferecem uma visão completa e dinâmica de uma organização, identificando e agrupando atividades dependentes executadas

por departamentos diferentes, muitas vezes separados hierárquica e funcionalmente, agregando valor ao cliente (Damij, 2007).

Uma característica encontrada na maioria dos processos, principalmente nos processos de negócios, é a sua inter-funcionalidade, ou seja, não são executados inteiramente em uma única área funcional (Kintschner, 2003).

Um processo de negócio é constituído de etapas, representadas por uma única unidade de trabalho, ou também denominada atividade ou transação, que pode ser executada completa ou parcialmente. Uma etapa de processo identifica unicamente um evento de negócio, e pode ser utilizada para auditoria e acompanhamento. Uma etapa de processo pode acessar e alterar o estado de uma ou várias entidades de negócio. Pessoas são partes integrantes de qualquer processo que não esteja totalmente automatizado. As etapas de processo possuem uma interface pela qual podem ser vistas e controladas por pessoas, os usuários do processo. A interface do processo com o usuário permite que este atue sobre atributos do processo, além de permitir que ações sejam tomadas, como iniciar, parar, suspender, restaurar, cancelar ou completar uma etapa de processo (Marshall, 2000).

As organizações necessitam se adaptar, de forma flexível e rápida, às constantes mudanças no ambiente empresarial do mundo globalizado. Seu desempenho competitivo depende de sua habilidade de atender exigências endógenas e exógenas de mudanças, tais como, evolução do ciclo de vida de produto, novos concorrentes ou alterações políticas e econômicas. Os processos de negócio devem evoluir de forma adequada e flexível para acompanhar as constantes e repetitivas transformações das organizações (Korthaus, 1998).

Os processos de negócio, geralmente, possuem um alto grau de complexidade e são realizados por uma seqüência mais detalhada de etapas. Um processo é

composto de um conjunto de etapas, ou atividades, organizadas de acordo com regras que definem a seqüência e a dependência entre elas (Marshall, 2000).

As organizações estão se focando cada vez mais em seus processos do negócio, resultando em um interesse crescente por melhores práticas de reengenharia. Dois problemas afloram da utilização destas melhores práticas: como encontrar, entre as melhores práticas, a que se adapte às finalidades da organização; e como assegurar que a melhor prática selecionada possua a mesma natureza do processo de negócio sob re-engenharia. Andersson et al (2005) propõe como solução deste problema utilizar processos de negócio padronizados (BPP - *Business Process Patterns*) e um método para realizar uma comparação formal entre os processos de negócio atual e os novos processos de negócio oriundos das melhores práticas.

Barros (2007) propõe um método baseado em processos de negócio padronizados (BPP - B*usiness Process Patterns*) e arcabouços (*frameworks*) que permite formal e logicamente a reutilização de conhecimento na criação e revisão dos processos de negócio das organizações.

A tecnologia da informação possui um grande potencial para auxiliar e automatizar partes dos processos de negócio (Akhavan et al, 2006). Entretanto, isto requer que as aplicações possam expressar e interpretar uma grande quantidade de informações e significados para estas informações. Ou seja, as aplicações que suportam os processos de negócio precisam ir além da simples expressão das informações, elas precisam interpretar de forma inteligente as informações para entender o seu significado (Rukanova et al, 2005).

Em seu trabalho, Rukanova (2005), apresenta uma alternativa para satisfazer este contexto operacional, utilizando padrões ou *Standards*, agrupados em três tipos

principais: padrões que definem o significado dos dados; os padrões que descrevem o que é comunicado em uma mensagem; e os padrões que definem as intenções da mensagem (Rukanova, 2005) e (Rukanova et al, 2005).

Sistemas de gerenciamento de processos de negócio ou BPMS[2] são ambientes associados à tecnologia da informação que oferecem ferramentas para a implementação de uma abordagem administrativa da gestão por processos, oferecendo recursos para a modelagem, integração, construção, execução e manutenção dos processos de negócio da organização (De Sordi & Spelta, 2007) (Shaw et al, 2007).

Em seu trabalho, "*Elements of a business process management system: theory and practice*" (Shaw et al, 2007), Shaw apresenta, baseado em uma extensa revisão da literatura acadêmica e comercial, os elementos comuns encontrados nos principais BMPS (Figura 1). A arquitetura em forma de pirâmide representa duas dimensões, uma tratando as questões sintáticas, semânticas e pragmáticas e, a outra, representando o lado tecnológico dos BPMS. A Tabela 1 apresenta uma breve descrição de cada um dos elementos que compõem um BPMS.

---

[2] *Business Process Management Systems*
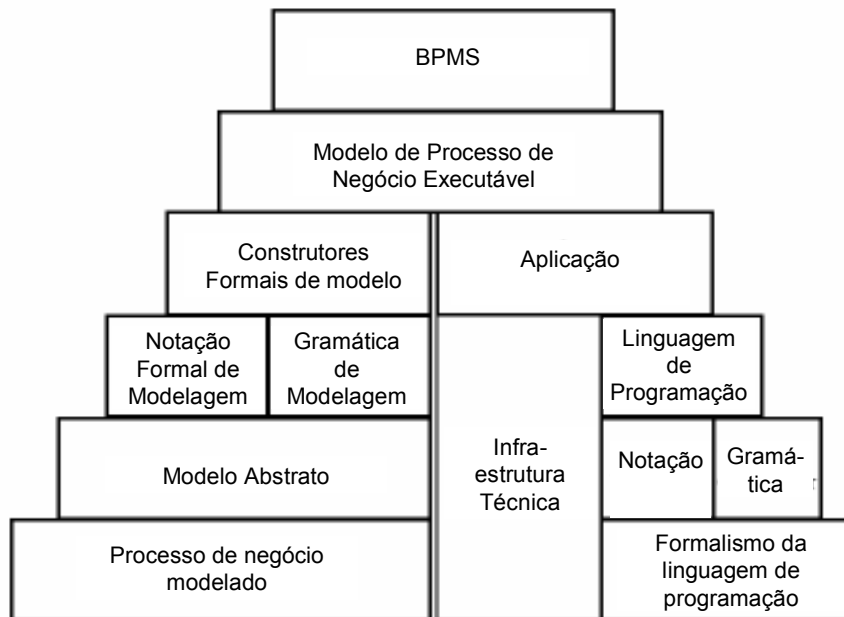
Figura 1 - Arquitetura BPMS proposta por Shaw et al (2007)

Tabela 1 - Descrição elementos da arquitetura BPMS (Shaw et al, 2007)

Elemento	Descrição
BPMS	**Ambiente para modelagem, integração e execução destinado ao *design*, construção e manutenção de processos de negócio**
Modelo de processo de negócio executável	**Elemento que permite a execução dos modelos de processo de negócio**
Construtores formais de modelo	**Elementos que compõem um modelo executável**
Notação formal de modelagem	**Conjunto de símbolos usados para representar os construtores formais de modelo**
Gramática de modelagem	**Conjunto de regras que definem os construtores formais de modelo**
Modelo abstrato	**Abstração da realidade descrita pela gramática de modelagem**
Processo de negócio modelado	**Processo de negócio que está sendo tratado pelo BPMS**
Aplicação	**Software que implementa o BPMS**
Infra-estrutura técnica	**Sistema operacional e o hardware que suportam a aplicação**
Linguagem de programação	**Linguagem de programação utilizada para implementar a aplicação**
Notação da linguagem de programação	**Notação da linguagem de programação utilizada para implementar a aplicação**
Gramática da linguagem de programação	**Gramática da linguagem de programação utilizada para implementar a aplicação**
Formalismo da linguagem de programação	**Formalismo da linguagem de programação utilizada para implementar a aplicação que define capacidade de prova matemática de suas propriedades**

2.2.2 Modelagem de Processos de Negócio

Para Santos et al (2002), a engenharia de processos de negócio é fortemente suportada por modelos de processos destinados a uma melhor compreensão da

organização. A modelagem de processos de negócio possui um conjunto de finalidades básicas destinadas à representação, análise e melhoria da forma como o trabalho é realizado horizontalmente, orientado para produtos, clientes e mercados, nas organizações. A modelagem de processos de negócio oferece suporte para um conjunto de aplicações da engenharia de processos de negócio: redesenho de processo; análise e melhorias de processos; sistemas integrados de gestão; projeto de sistemas de informação; identificação, seleção e monitoração de indicadores de desempenho; análises organizacionais; gerência do conhecimento; *workflow* e gerência de documentos; organização de documentação técnica; *benchmarking* entre as formas de trabalho; integração organizacional através da uniformização de entendimentos sobre a forma de trabalho; modelos de negócios eletrônicos; e cadeia de suprimentos.

Um modelo de negócio é uma abstração do funcionamento do próprio negócio e deve possuir os seguintes componentes (Dias et al, 2006) (Marshall, 2000) (Korthaus, 1998):

- Os objetivos são os propósitos do negócio, ou simplesmente, os resultados que toda a organização deseja atingir.
- Os recursos constituem os objetos utilizados em um negócio, tais como pessoa, material, informação ou produto.
- Os processos constituem um conjunto de atividades estruturadas para que um produto, que pode ser um bem ou um serviço, possa ser gerado.
- As regras são declarações que restringem, derivam e fornecem condições de existência, representando o conhecimento do negócio.

A definição de padrões e práticas comuns à modelagem permite dirimir as questões específicas ou polêmicas inerentes a um determinado modelo, oferecendo

a todas as pessoas envolvidas no trabalho de modelagem de negócio uma forma assemelhada de trabalhar e descrever processo. Os padrões e as práticas comuns refletem diretamente na legibilidade, percepção da qualidade e homogeneidade no resultado obtido com a modelagem dos processos de negócio (Cameira & Caulliraux, 2000).

Na busca por padrões de modelagem, diversos métodos de modelagem de negócio foram desenvolvidos com o intuito de facilitar o processo de modelagem empresarial através de métodos pré-definidos que pudessem ser reutilizados. Os métodos de modelagem de negócio oferecem a engenharia de processos de negócio um referencial único, integrado e articulado para o tratamento da modelagem de negócio. Seguem alguns destes métodos (Santos et al, 2002) (Vicente, 2004):

- *Architecture of integrated information systems* – ARIS (IDS, 2007): este método tem como principal objetivo, através de modelos, estabelecer uma visão holística, integrada e homogênea de uma organização, permitindo o desenvolvimento de sistemas de informação mais aderentes ao negócio. Este método está fundamentado na integração dos processos de negócios da organização através do agrupamento dos modelos em cinco vistas interligadas: organização, função, dados, saída, e controle ou processo. São diversos modelos associados a cada vista, que se inter-relacionam, mostrando, de maneira consistente, desde a concepção e o entendimento estratégico da organização até os componentes de implementação dos sistemas, com destaque para os seguintes modelos: o Diagrama de Objetivos, o Organograma, a Cadeia de Valor Agregado, a Cadeia de Eventos Orientados a Processos e o Diagrama de Caso de Uso.

- *Integrated DEFinition* - IDEF (IDEF, 2007): O método IDEF começou a ser criado nos anos 70 pela força aérea dos Estados Unidos da América para o tratamento apenas de sistemas. O IDEF, com o passar dos anos, foi agregando novos métodos relacionados a diversos aspectos organizacionais e atualmente é composto por diversos sub-métodos: IDEF 0, relacionado à modelagem funcional; IDEF 1, para modelagem de informações; IDEF 2, desenvolvido para suportar simulações; IDEF 3, para descrição e captura de processos; IDEF 4, para modelagem orientada a objetos; e IDEF 5, para descrição de ontologias. O método IDEF 3 está diretamente relacionado à modelagem de processos, sendo composto por dois modelos denominados Modelos de Fluxo de Processo (*Process Flow*), que descreve como as coisas são feitas dentro de um processo de produção, e Modelo de Rede de Transição de Estado de Objeto (*Object State Transition Network* - OSTN), que descreve os eventos pelos quais um objeto passa em um determinado processo.
- *Computer Integrated Manufacturing Open System Architecture* - CIMOSA (Vernadat, 1996): introduziu o conceito de arquitetura de sistemas abertos para CIM (*Computer Integrated Manufacturing*), através da padronização de módulos, descritos em termos de sua função, informação, recurso e aspectos organizacionais, oferecendo uma arquitetura capaz de auxiliar as organizações a gerenciar mudanças e integrar suas facilidades e operações com o objetivo de obter preços, qualidade e prazos de entrega competitivos. CIMOSA é uma arquitetura consistente tanto para a modelagem como para a integração empresarial e possui 3 componentes principais: estrutura de

modelagem empresarial, a infra-estrutura de integração e ciclo de vida do sistema CIM.

Vicente (2004) analisa as técnicas de modelagem de processos de negócio e as técnicas de modelagem de sistemas e propõe um método de modelagem de processos e requisitos de negócio, baseada em *Unified Modeling Language* (UML), capaz de apoiar a especificação e o projeto de um sistema integrando os requisitos de negócio com os requisitos de sistemas. Nesse trabalho, Vicente apresenta os principais conceitos, metodologias, ferramentas, aplicações e ganhos da engenharia de processos de negócio, ressaltando a importância da integração e do alinhamento entre a estratégia concebida para um negócio, seus processos e os sistemas que o suportam.

Para Damij (2007), existem diversas técnicas de modelagem de processos de negócio que podem ser divididas em dois grupos básicos: técnicas baseadas em diagramas e técnicas baseadas em tabelas. Em seu trabalho, Damij (2007) apresenta e compara técnicas de modelagem de processos de negócio e conclui que as técnicas devem ser utilizadas em conjunto, começando por um tratamento tabular e continuando com um tratamento diagramático.

Salm (2003) trata a modelagem de processos de negócio utilizando uma linguagem de modelagem orientada a objetos, a *Unified Modeling Language* (UML), mostrando que os artefatos produzidos na etapa de *design* e construção são guiados por modelos criados na disciplina de modelagem de processos de negócio e concluindo que a utilização da UML na modelagem de processos de negócio implica

a busca de extensões[3] da UML que permitam o tratamento adequado das necessidades impostas pela disciplina de modelagem de negócio.

Para Vicente (2004), é necessário definir um método de modelagem de processos com foco no desenvolvimento em sistemas, mesclando as principais etapas de modelagem de processos de negócio com algumas das etapas de modelagem de sistemas de informação. Conclui que o processo de desenvolvimento de sistemas proposto pelo *Rational Unified Process* (RUP), como arcabouço para o desenvolvimento de sistemas de informação, permite verificar de que forma os diagramas da UML são utilizados para a modelagem de negócio dentro de um processo de desenvolvimento de um sistema de informação.

2.2.3 Modelagem de Negócio com UML

A *Unified Modeling Language* (UML) é uma linguagem gráfica que permite a visualização, especificação, construção e documentação dos artefatos que compõem um sistema de software. A UML é uma linguagem para modelagem, com vocabulário e regras originalmente focados na representação conceitual e física de um sistema de software, que pode ser utilizada para modelagem de negócio e de sistemas em geral (Booch et al, 1999) (OMG, 2007).

Existem diversos trabalhos (Korthaus, 1998) (Marshall, 2000) (Noran, 2000) (Baker, 2001) (Jackowski, 2003) (Salm, 2003) (Vicente, 2004), utilizando UML 1 como linguagem gráfica para a modelagem de negócio, tratando de forma

---

[3] A UML pode ser estendida ou adaptada para um método específico, uma organização ou um usuário, através de elementos de modelagem que permitem o tratamento visual especializado e definem como criar novas semânticas (Salm, 2003) (OMGUML, 2007).

abrangente esta questão, descrevendo os passos e os recursos a serem utilizados, e indicando os problemas, as dificuldades e as inconsistências encontradas.

Vicente (2004) conclui que a modelagem de negócios com os modelos da UML versão 1, apesar de apresentar vantagens de entendimento por parte dos analistas desenvolvedores de sistemas, apresenta limitações de entendimento por parte dos analistas de negócio, pois, quando comparados com as técnicas de modelagem de negócio tradicionais, os modelos da UML não representam de forma satisfatória alguns conceitos relacionados à modelagem de negócio.

Em seu trabalho Vicente (2004) utiliza a proposta de extensão da UML versão 1 de Eriksson & Penker (1999) apud Vicente (2004), destinada à modelagem de negócio, com uma proposta de arquitetura dividida em vistas suportadas por modelos e extensões da UML, onde é obtido um maior acoplamento com as necessidades da modelagem de processos de negócio e utilizando, principalmente, o diagrama de atividades, para representar conceitos inerentes aos conceitos de processos de negócio. Nesta proposta, as principais limitações encontradas são a falta de capacidade de abstração nos níveis de detalhamento e a limitação para representar sistemas de informação que suportem estruturas de processos de negócio mais complexas.

Desde a sua apresentação, em 1997, a UML vem se tornando o padrão para a modelagem orientada a objetos de projetos de desenvolvimento de software. A UML versão 2.0, ou UML 2, apresentada pela OMG (OMG, 2007) (OMGUML, 2007) em 2004, inclui 13 notações distintas de modelagem, variando desde diagramas de alto nível, como o diagrama de casos de uso, que descrevem as interações e os relacionamentos entre atores e as funções básicas de negócio, até diagramas de baixo nível, como os diagramas de objetos, que capturam instâncias de objetos

individuas de dados. As notações de modelagem podem ser divididas em três grupos (Russel et al, 2006) (OMGUML, 2007):

- Diagramas de estrutura, que capturam a estrutura de estática do sistema em diversos níveis de abstração, incluindo: o diagrama de classe, o diagrama de objeto, o diagrama de componentes, o diagrama composição de estrutura, o diagrama de pacotes e o diagrama de distribuição.
- Diagramas do comportamento, que fornecem uma abstração de alto nível para as funcionalidades do sistema, incluindo: o diagrama de caso de uso, diagrama da atividade e o diagrama de máquina de estado.
- Diagramas da interação, que fornecem uma descrição mais detalhada das funcionalidades do sistema, incluindo as interações entre os objetos, incluindo: o diagrama de seqüência, o diagrama de comunicação, o diagrama temporal e o diagrama de visão geral da interação.

UML 2 é uma linguagem independente de ferramenta ou método utilizado para o desenvolvimento de sistemas. Pode ser utilizado XMI (*XML*[4] *Metadata Interchange*) para a transferência dos modelos UML existentes no repositório de uma determinada ferramenta, para o repositório de outra ferramenta que ofereça o maior refinamento ou o detalhamento necessário para uma etapa posterior estabelecida no método de desenvolvimento. XMI é um padrão da OMG para troca de informações baseado em XML, onde o uso mais comum é na troca facilitada de

---

[4] XML (eXtensible Markup Language) é uma linguagem capaz de descrever diversos tipos de dados, onde o propósito principal é a facilidade de compartilhamento de informações através da Internet (W3CXML, 2008).

dados entre as ferramentas de modelagem baseadas em UML e os repositórios MOF[5] (OMGUML, 2007).

A UML 2.0 permite navegar entre diversos níveis de abstração e detalhes do sistema. Isto é, partindo de uma vista detalhada da aplicação chega-se ao ambiente onde a aplicação está sendo executada, ou até mesmo ao processo de negócio ou à regra de negócio que a aplicação está automatizando (OMGUML, 2007).

De acordo com Selic (2006), a UML 2.0 apresenta um conjunto de diferenças em relação às versões anteriores, as quais estão listadas a seguir:

- Maior precisão na definição da linguagem: necessário para oferecer maiores níveis de automação, inerentes ao desenvolvimento orientado a modelo (MDD *Model Driven Development*), que requerem a eliminação de ambigüidades e imprecisões nos modelos.
- Arquitetura altamente modular da linguagem: que permite que a linguagem possa ser aprendida e utilizada de forma gradual, de acordo com as exigências de sofisticação do domínio e do projeto.
- Novos recursos de modelagem: a maioria dos novos recursos de modelagem da UML 2 são extensões daqueles já existentes e que, geralmente, permitem a modelagem mais direta e escalável de complexas aplicações distribuídas. Para conseguir ser escalável, muitos dos conceitos de modelagem da UML 2 são recursivos, permitindo que sistemas muito complexos sejam

---

[5] MOF (*MetaObject Facility Specification*) é um ambiente padronizado onde os modelos podem ser exportados de uma aplicação, importados em outra, transportados pela internet, armazenados ou recuperados de um repositório, traduzido ou transformado para outro formato, e utilizado para gerar o código da aplicação. (OMGMOF, 2007)

representados mediante a utilização de composição e decomposição hierárquica.

- Suporte aprimorado para linguagens de domínio específico: mesmo que a UML 2 padrão possa ser utilizada sem especialização adicional, a UML 2 é essencialmente uma linguagem de uso geral básica, que pode derivar linguagens de domínio específico altamente especializadas, oferecendo vantagens na reutilização direta de ferramentas UML e da capacitação existente das equipes de desenvolvimento.

As versões iniciais da UML eram basicamente baseadas no paradigma de orientação a objetos do desenvolvimento de software. A UML 2 oferece novos recursos para modelagem baseados na visão funcional de comportamento e na evolução do formalismo do modelo de atividades, que permitem a utilização de UML 2 em outros domínios, como modelagem de processos de negócio e engenharia de sistemas. Na UML 2 o diagrama de atividades possui o fundamento semântico baseado em Redes de Petri, ao invés do fundamento mais restrito oferecido pelas máquinas de estado utilizadas nas versões anteriores da UML (Selic, 2006).

Para Russell et al (2006), o diagrama de atividades da UML 2.0 tem mérito como uma notação para modelagem de processos do negócio, entretanto não é apropriado para todos os aspectos deste tipo de modelagem. O diagrama de atividades oferece uma sustentação detalhada para o controle de fluxo e para a perspectiva de dados, permitindo a modelagem direta da maioria das construções necessárias para representar estas perspectivas. Entretanto, oferece uma sustentação extremamente limitada para a modelagem dos aspectos relacionados a recursos ou aspectos organizacionais dos processos de negócio, pois não permite capturar grande parte das construções básicas encontradas nos processos do

negócio, como por exemplo, a noção de interação com o ambiente operacional onde os processo funcionam. Cabe ressaltar que estas limitações também são encontradas em grande parte das demais ferramentas para modelagem de negócio.

2.3 PROCESSO UNIFICADO

Segundo Pressman (2006), o processo unificado é uma tentativa de utilizar os melhores recursos e características dos modelos convencionais de engenharia de software, reconhecendo a importância da comunicação com o cliente e da arquitetura de software, através de um fluxo de processo iterativo e incremental, baseado na utilização de uma linguagem gráfica de modelagem, *Unified Modeling Language* (UML).

Os métodos e linguagens de programação orientada a objetos ganharam uma grande audiência na comunidade de engenharia de software nas décadas de 1980 e 1990, onde surgiram diversos métodos de análise orientada a objetos e de projeto orientada a objetos (Pressman, 2006). Entretanto em 1967 os sistemas já eram modelados com um conjunto de blocos interconectados, iniciando um processo com cerca de três décadas de evolução e amadurecimento para o processo unificado, surgindo o O*bjectory Process* em 1985, sucedido pelo *Rational Objectory Process* 4.1 em 1997, que incorporou a UML e finalmente em 1998 o *Rational Unified Process* 5.0 (Jacobson et al, 1999).

O processo unificado é um processo de engenharia de software, formado por um conjunto de atividades que objetivam transformar os requisitos dos stakeholders em um sistema de software. O processo unificado é baseado em componentes, utiliza UML (Booch et al, 1999) (Gilleanes, 2004) como elemento básico para a apresentação de toda a documentação do sistema, é dirigido a casos de uso, além

de ser um processo centrado na evolução da arquitetura[6] e apresentar uma abordagem iterativa e incremental (Jacobson et al, 1999).

2.3.1 *Rational Unified Process* - RUP

O RUP é um arcabouço para a engenharia de software orientada a objetos, formado por um conjunto de atividades necessárias para transformar os requisitos dos *stakeholders* em um conjunto consistente de artefatos que representam um produto de software. O RUP utiliza um conjunto de disciplinas para atribuir tarefas e responsabilidades dentro de uma organização de desenvolvimento de software. Sua meta é garantir a produção de software de alta qualidade que atenda às necessidades dos usuários, dentro de um cronograma e um orçamento previsíveis (Kruchten, 2000) (Pressman, 2006) (IBMRational, 2008).

O RUP busca capturar as melhores práticas da engenharia de software, tais como (Kruchten, 2000):

- Desenvolvimento de software iterativo, que busca a divisão do projeto em vários projetos menores ou iterações
- Gerenciamento de requisitos, que busca de forma sistemática a eliciação, organização, comunicação e gerenciamento das mudanças dos requisitos do sistema, permitindo avaliar e estimar os impactos das alterações.
- Utilização de arquitetura baseada em componentes, que busca produzir e validar a arquitetura de software nas etapas iniciais do ciclo de desenvolvimento, além de privilegiar a reutilização de software.

[6] Architecture-centric

- Modelagem visual, que, baseada na UML, busca auxiliar na especificação, construção, documentação e comunicação dos elementos do projeto, permitindo exibição ou inibição de detalhes de acordo com a visão a ser oferecida do sistema.
- Verificação contínua de qualidade de software, que busca distribuir a responsabilidade da qualidade, tanto do projeto como do produto, por toda a equipe de desenvolvimento.
- Controle de mudanças, que busca, de forma sistemática, o gerenciamento das mudanças nos requisitos, no *design* e na implementação.

O RUP apresenta uma abordagem iterativa em espiral e incremental, em que o projeto é dividido em vários projetos ou iterações. Cada iteração possui um conjunto definido de objetivos, direcionando a equipe de desenvolvimento para a obtenção de artefatos que convergem para o produto final esperado pelos *stakeholders*, oferecendo uma estratégia de desenvolvimento na qual o sistema é construído adicionando-se mais funcionalidades a cada iteração. Cada iteração apresenta elementos associadas à gerência de requisitos, análise e *design*, implementação, integração e testes. O RUP oferece uma estrutura iterativa onde o projeto é organizado em quatro fases: iniciação, buscando compreender o que deve ser feito através da definição do escopo do sistema; elaboração, buscando compreender como fazer através da definição da arquitetura do sistema; construção, buscando o desenvolvimento de uma versão preliminar do produto; e transição, buscando a construção da versão final do produto a ser implantada (Kroll, 2001).

A abordagem utilizando um modelo em espiral (Boehm, 1988) permite a identificação dos riscos existentes no projeto de forma antecipada dentro do ciclo de vida, sendo possível tratar os riscos de forma eficiente e no momento adequado,

permitindo a redução de custos e uma estimativa mais precisa dos prazos de desenvolvimento (Kroll, 2001).

O RUP é um processo que compreende as atividades das várias fases do ciclo de desenvolvimento do software, começando pela atividade de modelagem de negócio, passando pelas atividades de requisitos, análise, *design*, implementação, testes e implantação (Figura 2). O RUP está estruturado em duas dimensões (Kruchten, 2000):

- O eixo horizontal representa o tempo e os aspectos do ciclo de vida do desenvolvimento. Esta dimensão representa o aspecto dinâmico do processo e é expressa em termos de ciclos, fases, iterações e marcos.
- O eixo vertical representa as disciplinas, que agrupam as atividades de maneira lógica, de acordo com a sua natureza. Esta dimensão representa o aspecto estático do processo de desenvolvimento e de como o processo é descrito em termos de componentes, disciplinas, atividades, fluxos de trabalho, artefatos e papéis.
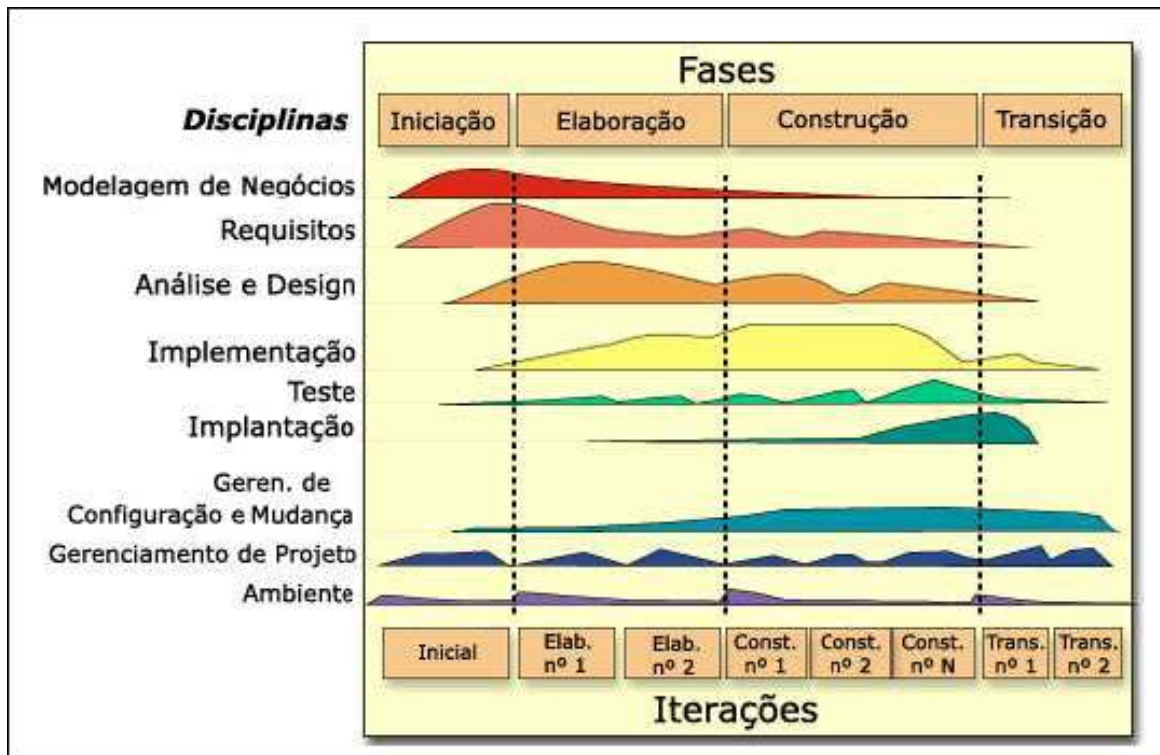
Figura 2 - Estrutura do RUP em duas dimensões (Kruchten, 2000)

Os requisitos do sistema são eliciados, não só para se encontrar a especificação do sistema, mas também para se criar uma representação comum para todos os integrantes da equipe de desenvolvimento e para os stakeholders. Geralmente, um sistema possui diversos tipos de usuários, onde cada tipo é representado por um ator. Um caso de uso descreve a forma como um ator interage com o sistema, através de uma seqüência de ações que o sistema executa para oferecer resultados aos atores. O conjunto formado por todos os atores e todos os casos de uso resulta no modelo de casos de uso do sistema (Jacobson et al, 1999). O RUP é dirigido pela utilização de casos de uso[7], onde a identificação de casos de uso e cenários típicos de utilização é a atividade que conduz todo o processo de desenvolvimento, desde a análise de requisitos até o teste do sistema final. A utilização dos casos de uso permite que a equipe de desenvolvimento trabalhe de

forma próxima aos requisitos durante o *design*, implementação e testes, garantindo que os requisitos serão atendidos.

O RUP contempla e reconhece em seu processo de desenvolvimento a modelagem do negócio como sendo uma disciplina fundamental para o desenvolvimento de sistemas e fornece técnicas de modelagem de negócio, mais especificamente os casos de uso de negócio, para representar processos de negócio. Entretanto esta proposta de representação apresenta limitações quanto à representação e modelagem dos fluxos dos processos de negócio e suas integrações, e também quanto ao alinhamento dos casos de uso identificados com os objetivos do negócio. Estas limitações podem ser contornadas integrando-se de forma mais consistente os conceitos de modelagem de processos de negócio com os conceitos de modelagem de sistemas, buscando a redução do tempo de transição entre os processos de negócio e o desenvolvimento dos sistemas e procurando uniformizar o entendimento geral entre analistas de negócio e de sistemas (Vicente, 2004).

2.4 ENGENHARIA DE REQUISITOS

No processo de desenvolvimento de sistemas é geralmente das etapas iniciais que decorre a maior parte dos problemas, no mínimo são aqueles mais dispendiosos e de maior impacto negativo. Estas etapas constituem o processo de engenharia de requisitos, que tem sido reconhecida como uma das mais importantes fases do processo de engenharia de software (Kotonya & Sommerville, 1998).

---

[7] Do termo em inglês *Use-Case Driven.*

A Figura 3 apresenta a relação entre o custo para reparar um problema de acordo com a fase em que este foi descoberto, dentro das etapas do ciclo de vida do desenvolvimento do sistema. O custo de um problema tratado quando o sistema está em produção é 500 vezes maior do que se fosse detectado e tratado na fase de requisitos (Carr, 2000).
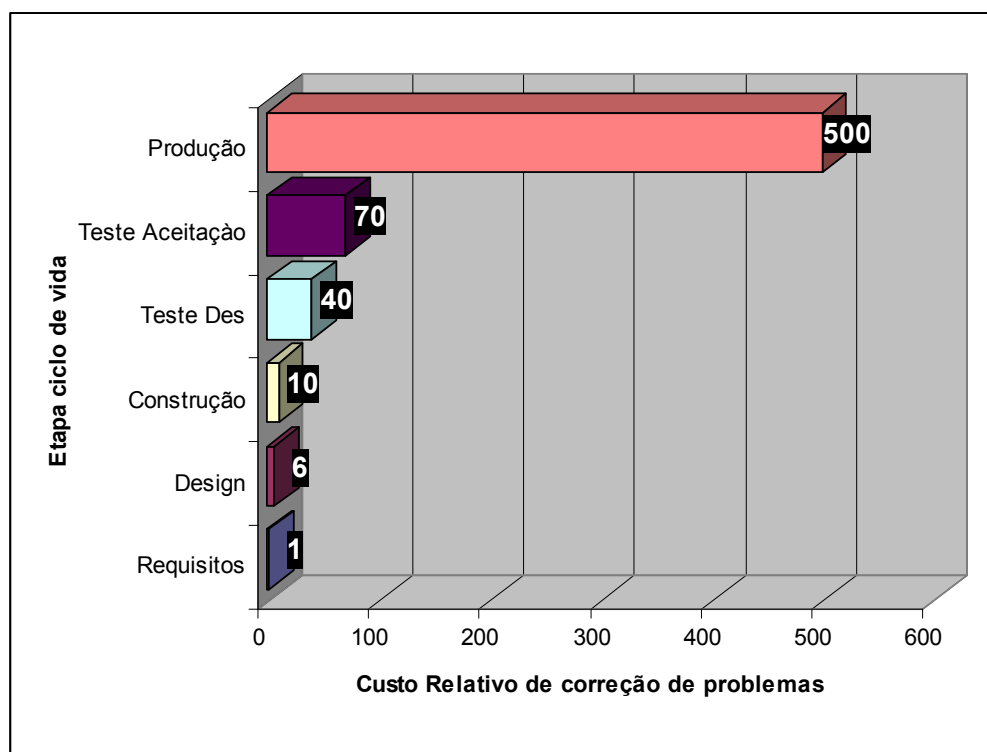
Figura 3 - Custo para corrigir problema no desenvolvimento de sistemas (Carr, 2000)

Leffingwell & Widrig (2000) destacam que a causa raiz do sucesso ou falha no desenvolvimento de um sistema está relacionado com os requisitos do sistema e, para fundamentar a afirmação, incluem em seu trabalho o estudo do *Standish Group* (1995), que indica que os três principais fatores causadores de problemas nos projetos de desenvolvimento de software são: falta de participação dos usuários (13%), especificações e requisitos incompletos (12%), e alteração nas especificações e requisitos (12%).

Leffingwell & Widrig (2000) destacam ainda a pesquisa realizada por ESPITI[8] em 1995, onde, através de 3800 respostas objetivando determinar os principais problemas encontrados no desenvolvimento de software, são identificados os dois principais problemas: especificação de requisitos e gerenciamento dos requisitos.

2.4.1 Definições

A engenharia de requisitos ajuda os engenheiros a compreenderem melhor o problema, através de um conjunto de tarefas que levam a um entendimento de qual será o impacto do sistema sobre o negócio, do que o *stakeholder* necessita e de como os usuários finais interagirão com o sistema (Pressman, 2006).

Segundo Kotonya & Sommerville (1998), engenharia de requisitos é o processo sistemático de eliciação, entendimento, análise e documentação dos requisitos, onde o termo engenharia implica a utilização de um processo prático, sistemático e repetitivo visando assegurar que os requisitos do sistema sejam completos, consistentes e relevantes.

Segundo Carr (2000), requisitos são as descrições das propriedades, dos atributos, dos serviços, das funções e dos comportamentos necessários em um produto para desempenhar os objetivos e finalidades do sistema. Um requisito define uma necessidade e não deve especificar uma solução de *design*.

Requisitos de sistemas definem o que o sistema necessita fazer e em quais circunstâncias deve operar. Requisitos são definidos nos estágios iniciais do desenvolvimento do sistema como sendo a especificação do que será implementado, definindo os serviços que um sistema deve prover e o conjunto de

---

[8] *European Software Process Improvement Training Iniciative*

restrições de operação que deve atender. A existência de diversos tipos de requisitos, para os mais vários tipos de sistemas, torna impossível descrever uma forma padronizada para escrever os requisitos e também definir a melhor forma de especificar requisitos. A descrição dos requisitos de um sistema deve incluir outras informações complementares como: informações de domínio, informações sobre padrões a serem seguidos e informações sobre interface com outros sistemas. Os requisitos não funcionais definem globalmente as qualidades ou atributos do sistema, sendo classificados em três tipos principais: requisitos do produto, requisitos do processo e requisitos externos (Kotonya & Sommerville, 98).

O termo *stakeholder* pode ser traduzido para o português como parte interessada ou interveniente, refere-se a todos os envolvidos em um processo: clientes, colaboradores, investidores, fornecedores, comunidade, organizações. O processo pode ser um projeto de caráter temporário ou o negócio duradouro de uma empresa. O sucesso de qualquer empreendimento depende da participação de *stakeholders* sendo necessário assegurar o atendimento à suas expectativas e necessidades (Kerzner, 2001).

Para Leffingwell & Widrig (2000) *stakeholder* é qualquer um que possa ser materialmente afetado pela implementação de um novo sistema ou de uma nova aplicação. Propõem um conjunto de questões para auxiliar no processo de definição dos *stakeholders* (Tabela 2).

Tabela 2 - Questões para definição dos *stakeholders* (Leffingwell & Widrig, 2000)

Quem são os usuários do sistema?
Quem é o cliente que está comprando o sistema?
Quem será afetado pelas saídas que o sistema produzirá?
Quem irá avaliar e aprovar o sistema quando for entregue e colocado em produção?

Existe outro usuário interno ou externo do sistema que necessite ser endereçado?
Quem irá manter o novo sistema?
Estes são todos?

Segundo Kotonya & Sommerville (1998), os *stakeholders* do sistema são as pessoas ou organizações que serão afetadas pelo sistema e que influenciam de forma direta ou indireta os requisitos dele. Os *stakeholders* incluem os usuários finais do sistema, gerentes, engenheiros responsáveis pelo desenvolvimento e manutenção do sistema, clientes da organização que utilizaram os serviços do sistema, organizações externas destinadas a certificações e regulamentações, entre outros. É fundamental identificar o grau de importância de cada *stakeholder* para o sistema e descobrir seus requisitos.

Para Kotonya & Sommerville (1998), o documento de requisitos é uma definição oficial dos requisitos do sistema para os clientes, usuários finais e desenvolvedores.

Carr (2000) define documento de requisitos como todo documento que descreve as propriedades, os atributos, os serviços, as funções e os comportamentos necessários para realizar os objetivos e as finalidades do sistema.

2.4.2 Processo de Engenharia de Requisitos

O processo de engenharia de requisitos é formado por um conjunto de atividades destinado a derivar, validar e manter o documento de requisitos do sistema. O processo apresenta um conjunto de entradas e um conjunto de saídas, representados na Figura 4 (Kotonya & Sommerville, 98):

- Informações de sistemas existentes (Entrada): informações associadas a funcionalidades de sistemas existentes sendo substituídos ou demais sistemas que irão interagir com o novo sistema.
- Necessidades dos *stakeholders* (entrada): descrição de quais as necessidades dos *stakeholders* devem ser atendidas pelo novo sistema.
- Padrões organizacionais (entrada): padrões utilizados pela organização associados a práticas de desenvolvimento, qualidade, gerenciamento e outros.
- Informações de domínio (entrada): Informações gerais quanto ao domínio onde o sistema será implantado.
- Regulamentações (entradas): regulamentações externas de segurança, saúde, governamental ou ambiental, aplicáveis ao desenvolvimento do novo sistema.
- Requisitos acordados (saída): descrição dos requisitos do sistema entendidos e aprovados pelos *stakeholders*.
- Especificação do sistema (saída): especificação detalhada das funcionalidades do sistema.
- Modelos do sistema (saída): conjunto de modelos que descrevem o sistema de diferentes perspectivas (modelo de fluxo de dados, modelo de objetos, modelo de processo, e outros).
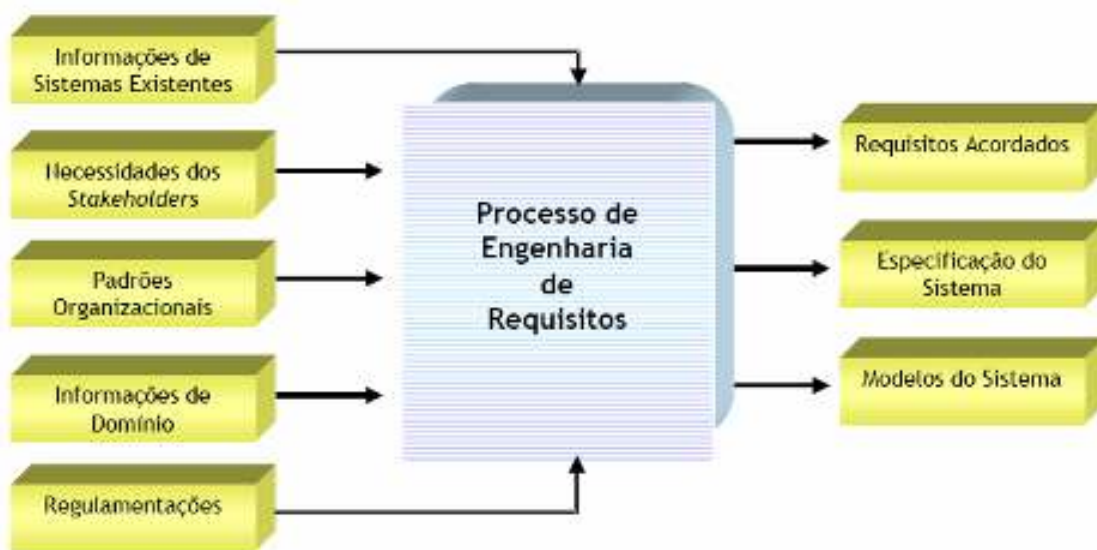
Figura 4 - Entradas e saídas do processo de engenharia de software (Kotonya & Sommerville, 98)

Para Pressman (2006), o processo de engenharia de requisitos é realizado através da execução de sete funções distintas: concepção, levantamento, elaboração, negociação, especificação, validação e gestão. Algumas destas funções ocorrem em paralelo e são adaptadas às necessidades do projeto, entretanto todas tem o objetivo de definir as necessidades dos *stakeholders*, estabelecendo uma fundação sólida de entendimento comum para o projeto e a construção do sistema.

Nuseibeh & Easterbrook (2000) propõem um conjunto básico de cinco atividades para o processo de engenharia de requisitos: eliciação, análise e modelagem, comunicação, concordância e evolução dos requisitos.

Existem diversas maneiras de organizar as atividades dentro do processo de engenharia de requisitos, dependendo do tipo de sistema a ser desenvolvido, da cultura da organizacional e do nível de experiência da equipe de desenvolvimento. Não existe um único processo de engenharia de requisitos que atenda a todas as organizações. Cada organização deve desenvolver o seu próprio processo, adequado as suas características e necessidades. Para Kotonya & Sommerville

(1998) o processo de engenharia de requisitos é composto pelas atividades de eliciação, análise, negociação e validação dos requisitos, organizadas conforme indicado na Figura 5.
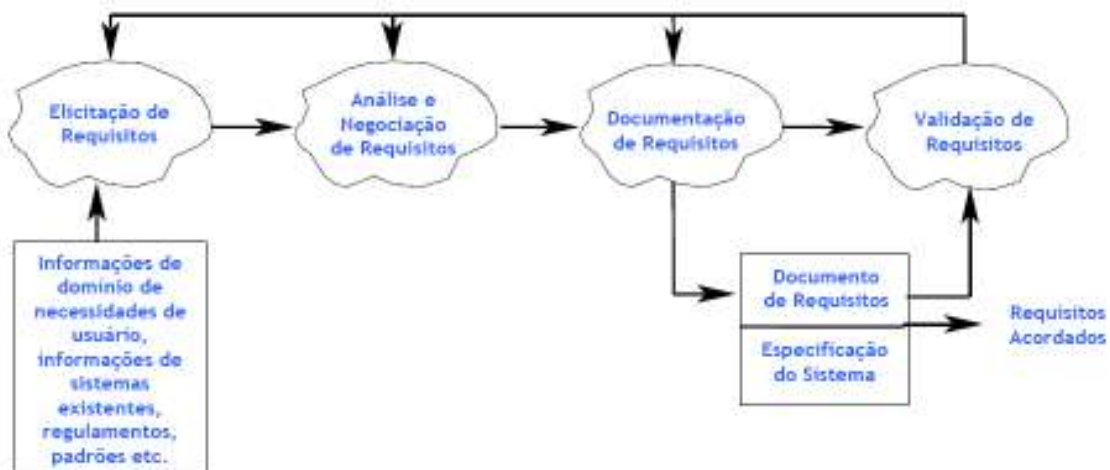
Figura 5 - Atividades do processo de engenharia de Requisitos (Kotonya & Sommerville, 98)

Entretanto o processo de engenharia de requisitos não é linear. Alterações nos requisitos são percebidas e devem ser tratadas de forma interativa durante o processo de engenharia de requisitos, como mostra a Figura 6 (Kotonya & Sommerville, 1998) (Boehm, 1988).
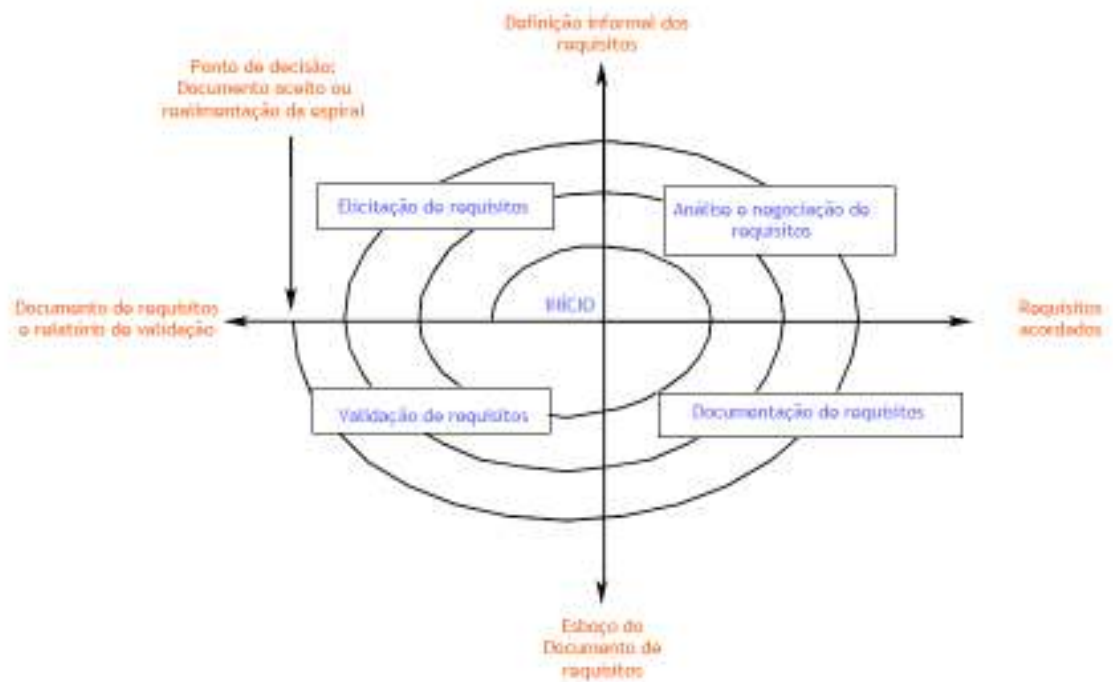
Figura 6 - Modelo espiral para os processos da engenharia de requisitos (Kotonya & Sommerville, 98)

Com o objetivo de completar o entendimento do processo de engenharia de requisitos, a Figura 7 apresenta o modelo definido por Dym & Little (2000) para o processo de engenharia de *design* de sistemas, onde a preocupação com requisitos se estende para as demais etapas do desenvolvimento do sistema.
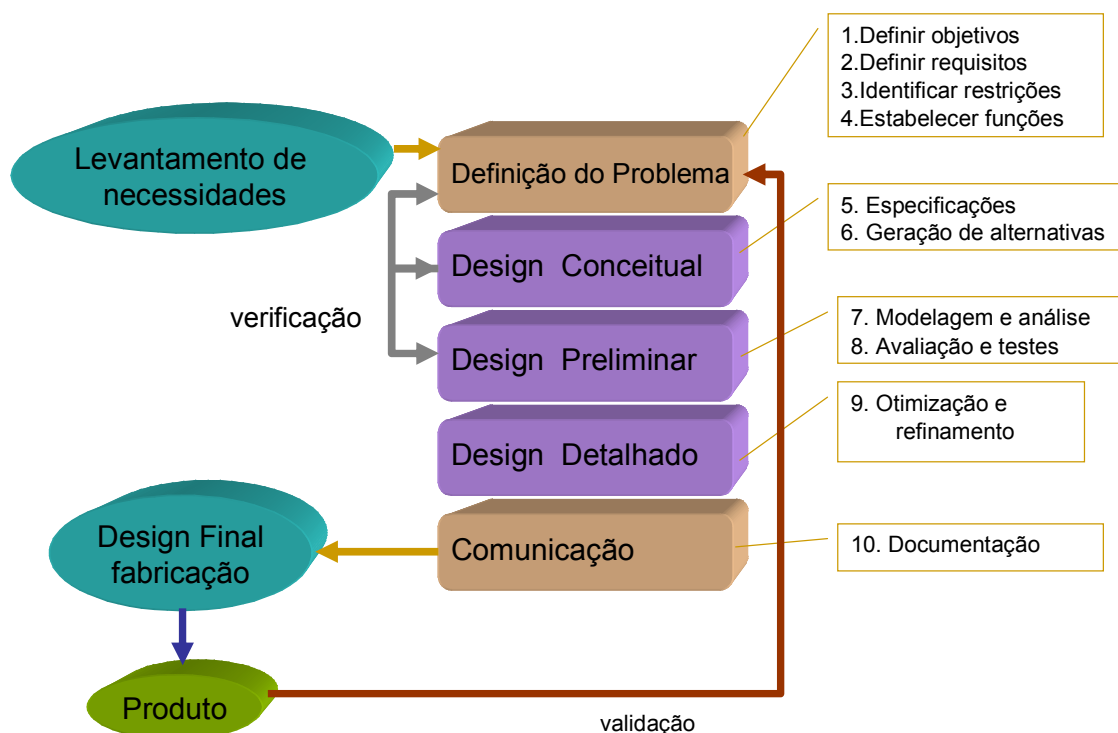
Figura 7 - Modelo do processo de *design* (Dym & Little, 2000)

Analisando-se as diversas abordagens para o tratamento de processos de engenharia de requisitos, observa-se que as atividades propostas são semelhantes e complementares, com algumas diferenças de nomenclatura e seqüência de utilização das atividades no processo. No entanto, será considerado para este trabalho, que a abordagem proposta por Kotonya & Sommerville (1998) fornece uma visão adequada para estas atividades.

2.4.3 Eliciação, análise e negociação de requisitos

A eliciação de requisitos compreende o conjunto de atividades que a equipe de desenvolvimento utiliza para descobrir as solicitações dos *stakeholders*, buscando traduzir as reais necessidades escondidas nestas solicitações. As necessidades, expectativas e recursos esperados do sistema serão levantados junto a cada

stakeholder. Também é necessário analisar os requisitos em relação a inconsistências e pontos incompletos, bem como negociar, solucionando conflitos, de forma que um conjunto de requisitos sejam acordados. O objetivo é delimitar um conjunto de requisitos documentados e priorizados que atendam às reais necessidades dos *stakeholders*.

A eliciação dos requisitos está associada ao processo de descoberta e análise do problema a ser resolvido, detectando o grau de importância de cada stakeholder e suas reais necessidades associadas ao problema, além de entender as restrições impostas ao sistema. Para está análise a eliciação de requisitos deve ser abordada por quatro dimensões (Kotonya & Sommerville, 1998):

- Entendimento do domínio da aplicação: compreensão geral da área onde o sistema é aplicado.
- Entendimento do problema: compreensão de detalhes do problema onde o sistema será utilizado, através da especialização do conhecimento geral estabelecido para o domínio do problema.
- Entendimento do negócio: compreensão de como o sistema interage, afeta e contribui para os objetivos de negócio da organização.
- Entendimento das necessidades e restrições: compreensão dos processos de trabalho que o sistema pretende suportar e às regras a que estará sujeito.

As atividades de eliciação, análise e negociação dos requisitos podem ser entendidas como processos interligados e interativos, representados por uma espiral de atividades que converge para a estabilização dos requisitos retratados no documento de requisitos (Kotonya & Sommerville, 1998).

O processo de eliciação de requisitos apresenta quatro atividades básicas: ajuste dos objetivos da organização, aquisição de conhecimentos básicos do

sistema, entendimento da organização e coleta de requisitos. A Figura 8 apresenta um modelo genérico para o processo de eliciação de requisitos (Kotonya & Sommerville, 1998).
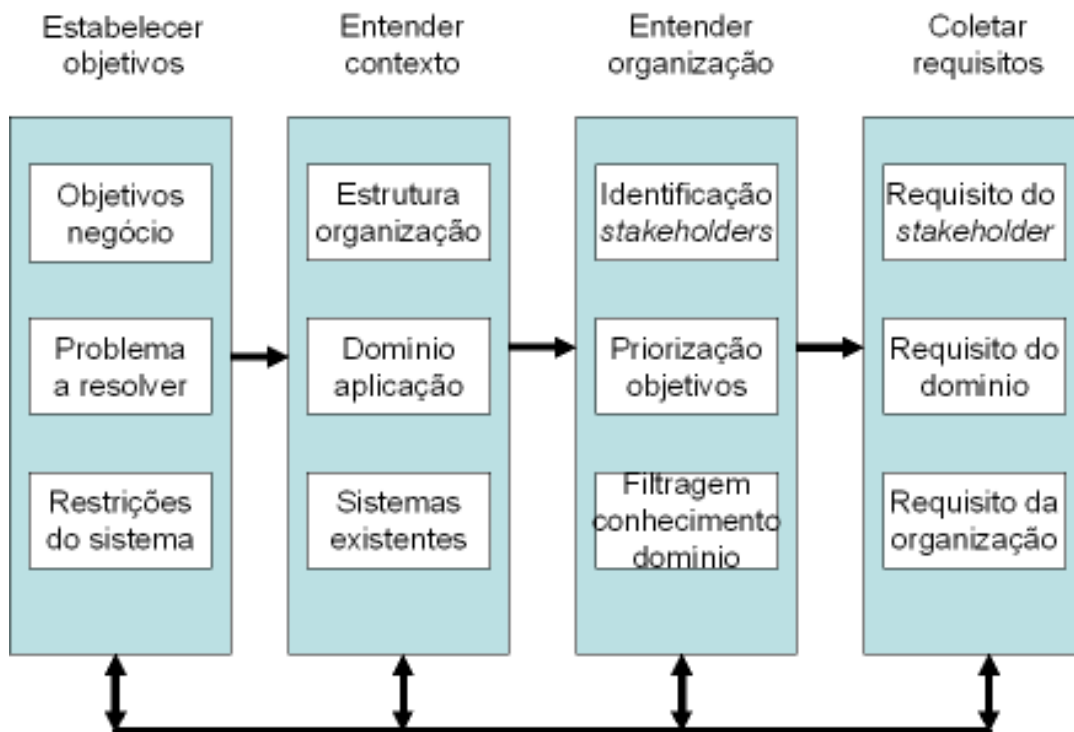
Figura 8 - Processo de eliciação de requisitos (Kotonya & Sommerville, 1998)

O documento preliminar de requisitos é refinado através da execução das atividades do processo de análise de requisitos. A Figura 9 apresenta as atividades de forma linear, entretanto estas atividades são intercaladas no tempo (Kotonya & Sommerville, 1998). As atividades de verificação realizadas no processo de análise são: verificação das necessidades, verificação de consistência e integridade e verificação de viabilidade.
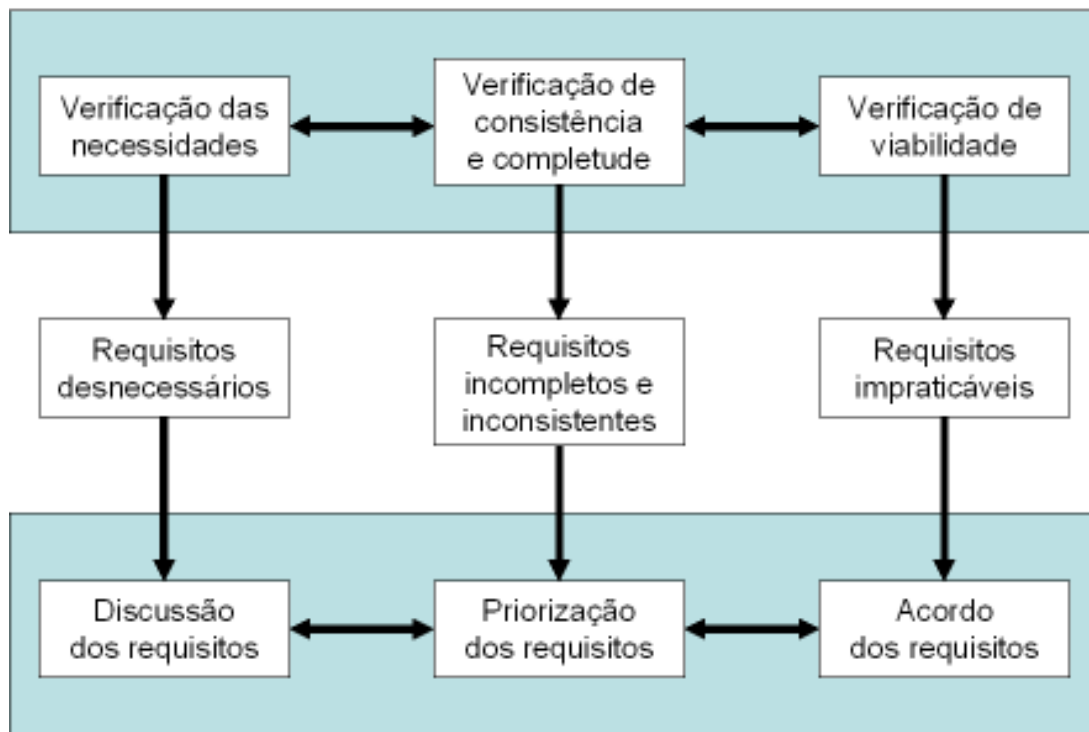
Figura 9 - Processo de análise e negociação dos requisitos (Kotonya & Sommerville, 1998)

A seleção da técnica adequada para a eliciação dos requisitos depende do tempo e dos recursos disponíveis à equipe de desenvolvimento e do tipo de sistema. As técnicas de eliciação podem ser agrupadas em classes (Nuseibeh & Easterbrook, 2000):

- Técnicas tradicionais: baseadas na utilização de técnicas genéricas para coleta de dados e informações como questionários, entrevistas e a análise da documentação existente.
- Eliciação em grupo: baseada no envolvimento dos *stakeholders*, explorando dinâmicas de grupo para favorecer o entendimento das necessidades. Inclui técnicas de *brainstorming* e reuniões de RAD e JAD[9] (Maiden & Rugg, 1996).

---

[9] RAD Rapid Application Development e JAD Joint Application Design

- Prototipação: geralmente é utilizada está técnica quando existem muitas incertezas quanto aos requisitos, ou quando é necessário que a realimentação do *stakeholder* seja antecipada. Um protótipo pode ser utilizado para apoiar outras técnicas de eliciação como fomentador de discussões e para esclarecimento de dúvidas de funcionamento.
- Dirigida a modelo: fornece um modelo específico para cada tipo de informação a ser tratada e utiliza este modelo para direcionar o processo de eliciação. Inclui métodos de modelagem baseados em objetivos, como KAOS (Van Lamsweerde et al, 1998) e I* (Alencar et al, 2000), e métodos de modelagem baseados em cenários (Maiden et al, 1999).
- Cognitiva: baseada em técnicas de aquisição de conhecimento destinadas a sistema baseados em conhecimento (Shaw & Gaines, 1996).
- Contextuais: alternativa para as técnicas tradicionais e cognitivas, baseada em técnicas etnográficas de observação participante e análise de conversação para busca de padrões (Viller & Sommerville, 1999).

2.4.4 Validação de requisitos

Sendo o estágio final da engenharia de requisitos, a atividade ou processo de validação de requisitos tem como objetivo estabelecer que os requisitos estejam definidos de forma correta, consistente, íntegra e precisa. O processo de validação dos requisitos envolve os *stakeholders*, os engenheiros de requisitos e a equipe de *design*, através da análise dos requisitos documentados, buscando problemas, omissões e ambigüidades.

O principal problema na validação dos requisitos é que não há uma referência documentada para ser utilizada como base na validação do documento de

requisitos. A validação de requisitos é um processo longo, envolvendo um grupo heterogêneo de pessoas que estão tratando um extenso e complexo documento.

O tratamento da validação de requisitos como um processo permite definir quais são as entradas e saídas do processo, bem como estabelecer um conjunto de atividades a serem realizadas dentro do processo (Kotonya & Sommerville, 98). Seguem as entradas e saídas associadas ao processo de validação de requisitos:

- Documento de requisitos (entrada): versão completa do documento obtida do processo de eliciação, análise e negociação dos requisitos e acordado com os stakeholders. O documento é organizado e formatado de acordo com os padrões da organização.
- Padrões organizacionais (entrada): o processo de validação deve estar em conformidade com as regras e padrões da organização onde o sistema será implantado.
- Conhecimento organizacional (entrada): as pessoas envolvidas com a validação dos requisitos devem conhecer a organização, suas particularidades, terminologias, práticas, capacitações individuais e coletivas.
- Lista de problemas (saída): lista de problemas obtida como resultado do tratamento das entradas do processo de validação.
- Lista de ações (saída): lista de ações obtida em resposta aos problemas encontrados nos requisitos durante o processo de validação.

As atividades do processo podem ser organizadas de acordo com diversas técnicas, que podem ser aplicadas individualmente ou de forma conjunta, dependendo da necessidade. A atividade de validação de requisitos é a última atividade para certificar-se de que o documento de requisitos satisfaz as

necessidades dos *stakeholders*. Várias são as técnicas empregadas (Kotonya & Sommerville, 98):

- Revisões dos requisitos: é uma das alternativas mais utilizadas no processo de validação. Consiste na revisão dos requisitos por um grupo de pessoas que analisam, discutem e propõem alternativas para solucionar os eventuais problemas encontrados. A partir dessas descobertas, pode-se adotar medidas que solucionem os conflitos. Problemas típicos encontrados nas revisões incluem: requisitos incompletos, ambigüidade, redundância, falha na rastreabilidade e não obediência a padrões da organização.
- Validação de modelos: o documento de requisitos pode estar descrito em linguagem natural e também em diagramas ou modelos. Os modelos normalmente desenvolvidos são: modelos de fluxo de dados, modelo de objetos, modelos de eventos e modelos de dados. Estes modelos são validados para demonstrar que cada um é consistente, que os modelos do sistema estão consistentes entre si e que representam precisamente os requisitos dos *stakeholders*.
- Prototipação: A técnica de prototipação geralmente é utilizada para auxiliar as atividades de eliciação e análise de requisitos, entretanto, pode ser empregada também na validação dos requisitos, onde o protótipo pode ser aperfeiçoado com as informações do documento de requisitos final e utilizado para a validação junto aos *stakeholders*. Os protótipos são desenvolvidos com o intuito de permitir uma melhor representação dos requisitos de um sistema, permitindo que o usuário tenha uma idéia antecipada de como o sistema funcionará.

- Teste de requisitos: um atributo desejável para um requisito é que este possa se testado. Deve ser possível definir um ou mais testes que, quando aplicados ao sistema final desenvolvido, possam claramente demonstrar que o requisito foi atendido pelo sistema. A dificuldade em definir e propor casos de teste adequados pode indicar problemas na definição dos requisitos. A proposição de casos de teste atua como um instrumento de validação identificando problemas nos requisitos.

2.4.5 Gerenciamento de requisitos

Leffingwell & Widrig (2000) propõem um processo para gerenciamento da engenharia de requisitos baseado na formação adequada das equipes de desenvolvimento, que atuam de forma coordenada, em que a base apresenta os requisitos do sistema, sustentando as características que atendem às necessidades dos *stakeholders*. As equipes são agrupadas por capacitação para realizar as seguintes atividades:

1. Análise do problema: que compreende a análise do problema, modelagem de negócio e engenharia de sistemas.
2. Entendimento das necessidades dos *stakeholders*: onde ocorre a eliciação dos requisitos, com a utilização de diversas técnicas para endereçar os problemas e entender as necessidades dos *stakeholders*.
3. Definição do sistema: trata da definição de um sistema que atenda às necessidades dos *stakeholders*.
4. Gerenciamento do escopo: tratando o gerenciamento do escopo do projeto que está representado na especificação dos requisitos.

5. Refinamento da definição do sistema: trata da elaboração dos requisitos para obtenção de um grau de detalhe adequado para as etapas de *design* e implementação.
6. Construção do sistema: trata da construção do sistema que atenda aos requisitos especificados, permitindo validação junto aos *stakeholders* e garantindo a rastreabilidade dos requisitos.

Para Pressman (2006) o gerenciamento de requisitos é um conjunto de atividades que auxiliam a equipe de desenvolvimento a identificar, controlar e rastrear requisitos e alterações em requisitos, dentro do ciclo de vida do projeto. Após a identificação os requisitos são incluídos em tabelas de rastreamento que os relacionam a um ou mais aspectos do sistema ou ambiente, como características dos requisitos, fonte de obtenção do requisito, dependência com os demais requisitos, associação com sub-sistemas e relação com interfaces do sistema.

Para Kotonya & Sommerville (1998) o gerenciamento de requisitos é um processo de gestão das mudanças ocorridas nos requisitos do sistema, decorrentes da melhor compreensão das reais necessidades, por parte dos *stakeholders*, dentro da evolução do ciclo de vida do projeto e das alterações no próprio ambiente onde o sistema está inserido. O processo de gerenciamento de requisitos possui três pontos principais:

- Gerenciar os requisitos acordados.
- Gerenciar as relações existentes entre os requisitos.
- Gerenciar as dependências entre o documento de requisitos e os demais documentos produzidos durante o ciclo de vida do projeto.

As mudanças nos requisitos são inevitáveis, pois tanto o entendimento do sistema por parte dos *stakeholders*, como os ambientes político, organizacional e

técnico onde o sistema está inserido, evoluem com o andamento do desenvolvimento do sistema. Neste contexto, ações de gerenciamento devem ser tomadas não somente durante o processo de engenharia de requisitos, mas durante todo o ciclo de vida do sistema. Cada requisito deve ser identificado unicamente, oferecendo recursos para o tratamento da rastreabilidade dos requisitos e permitindo avaliar o impacto decorrente das mudanças em vários níveis. Quando o volume de requisitos é muito grande, o que ocorre no desenvolvimento de grandes sistemas, a utilização de ferramentas de controle e utilização de banco de dados passa a ser indispensável, tanto para armazenar os requisitos como para controlar o relacionamento entre eles (Kotonya & Sommerville, 1998).

2.4.6 Métodos para engenharia de requisitos

A engenharia de requisitos é aplicada geralmente em um contexto de relações e atividades humanas, pressupondo a utilização de um sistema computadorizado e que possivelmente causará mudanças nas atividades por ele suportadas. Conseqüentemente, a engenharia de requisitos deve ser sensível à forma como as pessoas estão inseridas no mundo e como o trabalho afeta suas ações. A engenharia de requisitos utiliza as ciências cognitivas e sociais para fornecer fundamento teórico e técnicas práticas para eliciar e modelar requisitos (Nuseibeh & Easterbrook, 2000):

- Psicologia cognitiva fornece o entendimento das dificuldades que as pessoas podem ter em descrever as suas necessidades.
- Antropologia fornece uma aproximação metodológica para observar as atividades humanas auxiliando na compreensão de como os sistemas podem ajudar ou atrapalhar as pessoas.

- Sociologia fornece uma compreensão das mudanças políticas e culturais causadas pela informatização. A introdução de um sistema novo muda a natureza do trabalho realizado dentro de uma organização, pode afetar a estrutura e os meios de comunicação dentro dessa organização, podendo até alterar os objetivos originais das organizações.
- Lingüística é fundamental já que a engenharia de requisitos é sustentada por comunicação. As análises lingüísticas mudaram a forma como a linguagem natural é utilizada nas especificações, buscando evitar ambigüidades e melhorar a compreensão.

Os métodos de engenharia de requisitos são utilizados para auxiliar o processo de eliciação, estruturação e formatação dos requisitos do sistema, sendo uma forma sistemática para a produção de modelos do sistema (Kotonya & Sommerville, 1998).

A modelagem de requisitos auxilia no entendimento da informação, da função e do comportamento de um sistema, tornando a análise de requisitos mais fácil e mais sistemática; fornece o principal instrumento para a revisão de requisitos, influenciando na determinação da completude, consistência e precisão da especificação; e é a referência para as demais etapas do projeto de desenvolvimento do sistema (Alencar, 1999).

Segundo Kotonya & Sommerville (1998) não existe um método ideal para a engenharia de requisitos, pois poucos, ou nenhum, possuem todos os atributos necessários. Os principais métodos e técnicas de engenharia de requisitos são: modelagem de fluxo de dados, modelagem semântica, modelagem orientada a objetos, modelagem formal, SADT (*Structured Analysis and Design Technique*), CORE (*Controlled Requirement Expression*), VOSE (*Viewpoint-oriented System Engineering*) e VORD (*Viewpoint-oriented System Definition*).

2.5 SYSML

SysML (*Systems Modeling Language*) (OMGSysML, 2006) é uma linguagem gráfica de modelagem de sistemas para uso geral. SysML permite a modelagem de sistemas e arquiteturas de produtos, bem como seus comportamentos e funcionalidades, baseada na experiência adquirida na disciplina de engenharia de software em construir arquitetura de software em UML (Balmelli, 2006).

SysML é o resultado de uma iniciativa comum da OMG (*Object Management Group*) (OMG, 2007) e do INCOSE (*The International Council on Systems Engine*ering) (INCOSE, 2007) para criar uma linguagem unificada de modelagem de sistemas que atendesse à requisição de proposta da OMG, "*UML for Systems Engineering RFP*" (OMGRFP, 2003) doravante denominada apenas RFP[10], cujo objetivo foi a submissão de uma especificação de personalização da UML para atender as necessidades da engenharia de sistemas.

Baseada em uma linguagem gráfica, SysML suporta a especificação, a análise, o projeto, a verificação e a validação de um grande conjunto de sistemas complexos e heterogêneos, não necessariamente baseados em software, e suportando a modelagem de sistemas integrando hardware, software, dados, pessoas, procedimentos, processos e infra-estrutura (Vanderperren, 2005) (OMGSysML, 2006).

As empresas que desenvolvem sistemas estão sendo pressionadas pelo mercado e pelos concorrentes a melhorar a eficiência de seus projetos e sistemas de produção. Analisando o ciclo de vida do produto, observa-se uma falta de eficiência na fase conceitual, durante a qual são decididas as questões associadas à

[10] Request for Proposal ou solicitação de proposta

arquitetura funcional. A fase conceitual segue a transformação das necessidades dos clientes em função do produto e casos de uso, e precede o estágio de *design* destas funções através das diversas disciplinas da engenharia, como elétrica, mecânica e software. SysML foi projetada para mitigar os riscos e os desafios encontrados durante a fase conceitual do produto (Balmelli, 2006):

- Tratando de forma mais eficiente a realização dos requisitos em funções a serem desempenhadas pelo produto;
- Fornecendo uma representação formal dos conceitos do sistema e uma sustentação adequada para a tomada de decisão quanto viabilidade do produto.

Semelhante ao obtido com UML quanto à unificação de uma linguagem para modelagem de software, SysML pretende unificar as diversas linguagens utilizadas pelos engenheiros de sistemas para a modelagem de sistemas (OMGSysML, 2006).

SysML é uma extensão da UML versão 2.0. Os diagramas de Máquina de Estado (*State Machines*), de Interação (*Interactions*) e de Casos de Uso (*Use Cases*) são reutilizados da UML sem modificações, os diagramas de Atividade (*Activities*) e de Blocos (*Blocks*) são reutilizados de UML e estendidos em SysML, e finalmente, os diagramas de Requisitos (*Requirements*), de Parâmetros (*Parametrics*) e de Alocação (*Allocation*) são novos e disponíveis apenas em SysML (Figura 10) (Balmelli, 2006).
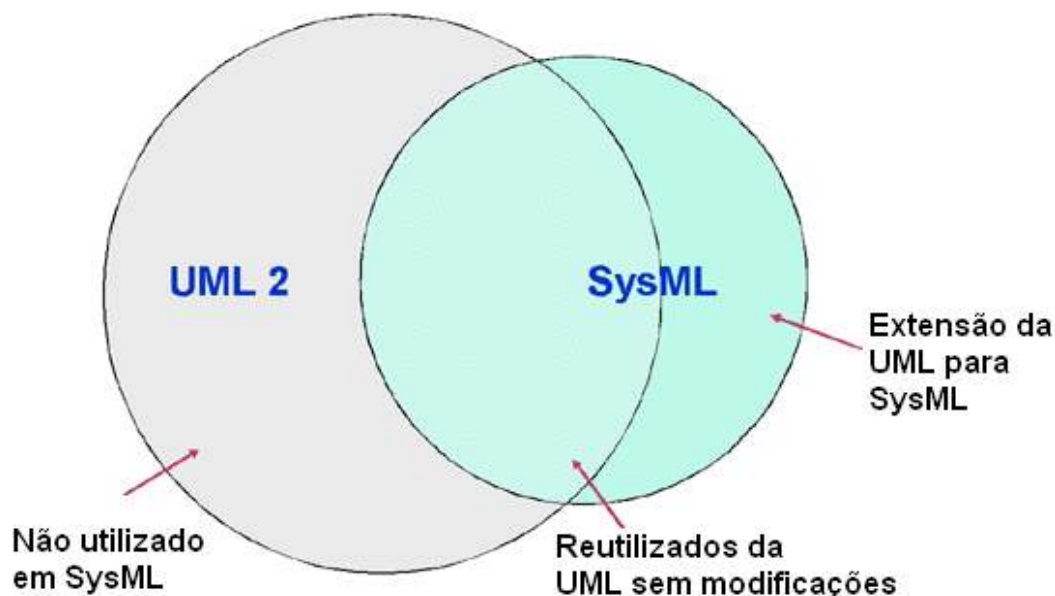
Figura 10 - Comparação entre UML 2.0 e SysML 1.0 (OMGSysML, 2006)

SysML utiliza XMI[11] (OMG XML Metadata Interchange) (OMG, 2007) para trocar dados de modelagem entre ferramentas e também é compatível com padrão ISO 10303-233 (ISO, 07) destinado ao intercâmbio dos dados na Engenharia de Sistemas (OMGSysML, 2006) (Peak, 2007a).

2.5.1 UML for System Engineering RFP

Sendo a SysML resultado do atendimento de um processo de requisição de proposta, neste trabalho serão abordados alguns itens do documento RFP, visando uma melhor compreensão dos conceitos associados à linguagem SysML.

O processo completo de uma RFP – *Request for Proposal* ou Requisição de Proposta permite avaliar e comparar propostas oferecidas pelos fornecedores, através do envio, a alguns fornecedores selecionados, de um documento que defina precisamente quais atividades devem ser desempenhadas pelo sistema, produto ou

---

[11] XMI (XML Metadata Interchange)

serviço ofertado, todos os prazos envolvidos, a equipe de trabalho necessária, o grau de detalhamento a ser utilizado e, principalmente, as conseqüências de um eminente fracasso. As propostas recebidas dos fornecedores são validadas, comparadas e avaliadas, permitindo que a empresa selecione a proposta mais adequada, sempre através de um método objetivo e preciso (Cundiff, 1997) (Bosik, 1997).

A RFP solicita o desenvolvimento de uma linguagem baseada em UML e que atenda às necessidades da Engenharia de Sistemas (SE), suportando a modelagem de diversos tipos de sistemas complexos e compostos pelos mais variados elementos como: hardware, software, dados, pessoas, procedimentos e infra-estrutura.

A RFP utiliza o termo “UML para Engenharia de Sistema” para referenciar o produto obtido com as respostas a RFP e que será substituído neste trabalho apenas pelo termo SysML, visando a simplificar o texto e facilitar o entendimento.

A RFP define que a SysML deve suportar a análise, a especificação, o projeto e a verificação de sistemas complexos, permitindo (OMGRFP, 03):

- capturar a informação dos sistemas de uma maneira precisa e eficiente, permitindo a integração e reutilização em larga escala;
- analisar e avaliar o sistema que está sendo especificado, para identificar e resolver os exigências de requisitos e *design* do sistema e para suportar sua evolução;
- comunicar as informações do sistema de forma correta e consistente aos vários tipos de stakeholders e participantes do projeto.

A SysML deve suportar a troca de informações entre as fases de análise, especificação, *design* e verificação existentes em um projeto de desenvolvimento de

sistemas, usando uma notação padronizada e uma semântica precisa e consistente, melhorando a comunicação entre os participantes do processo de desenvolvimento do sistema e promovendo a interoperabilidade entre as ferramentas que suportam este processo. A nova linguagem pode também estabelecer uma estrutura padrão de modelagem, podendo ser personalizada para necessidades específicas (OMGRFP, 03).

A RFP define que a nova linguagem deve ser obtida através dos mecanismos de extensão providos pela UML, combinados com os mecanismos da MOF (*Meta-Object Family*) (OMGMOF, 2007). A nova linguagem deve manter o relacionamento e a compatibilidade com as evoluções das especificações da OMG.

A RFP está estruturada em seis capítulos e dois apêndices, dos quais pode ser destacado o capítulo dois, que estabelece contexto arquitetural baseado na MDA (*Model Driven Architecture*) da OMG (OMGMDA, 2007) (Harmon, 2004). O capítulo seis, “Requisitos Específicos da Proposta”, também merece destaque, pois estabelece diretamente os requisitos a serem atendidos pela SysML 1.0. Os requisitos obrigatórios da RFP representam o núcleo básico das necessidades da Engenharia de Sistemas e devem ser atendidos na versão inicial da nova linguagem, SysML 1.0. Os requisitos obrigatórios estão organizados em elementos para modelagem de sistemas e foram agrupados nas seguintes categorias: estrutura, comportamento, propriedades, requisitos, verificação e complementares (OMGRFP, 2003).

2.5.2 Diagramas SysML

SysML utiliza uma variedade de diagramas (Figura 11) semanticamente fundamentados e que permitem a representação de sistemas complexos. SysML

modifica algumas construções existentes em UML e adiciona outras específicas para o tratamento das necessidades da engenharia de sistemas (OMG, 2007). Os diagramas estão agrupados em três categorias: estrutura, comportamento e requisitos.
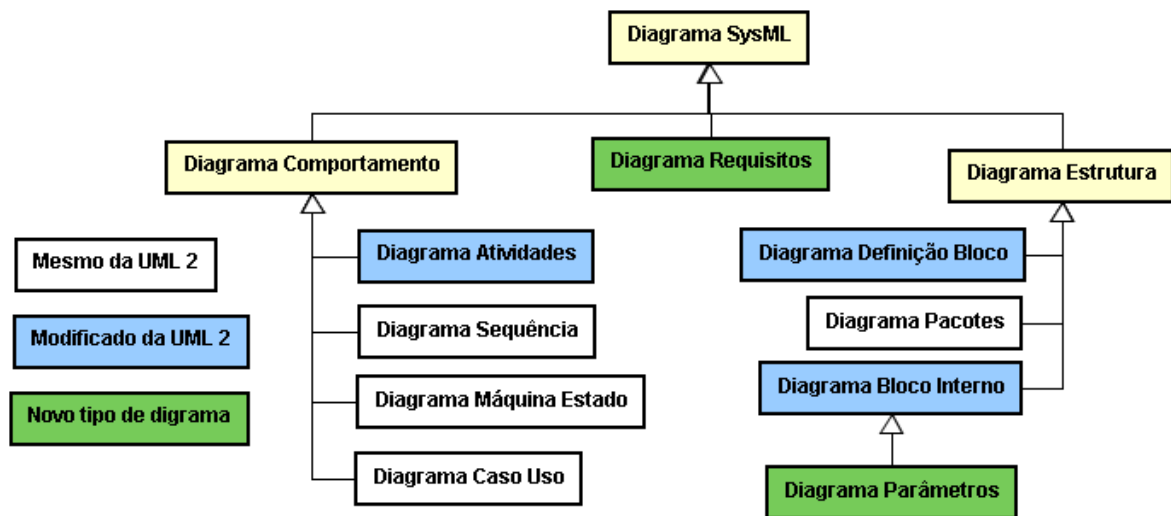
Figura 11 - Tipos de Diagramas da SysML (OMGSysML, 2006)

O bloco é o elemento básico para a representação da estrutura do sistema em SysML e pode ser utilizado para representar hardware, software, instalações, pessoas, ou outro elemento do sistema. A estrutura do sistema é representada pelo diagramas de definição de bloco e pelo diagrama de bloco interno. O diagrama de definição de bloco descreve a hierarquia do sistema e a classificação dos sistemas e componentes.

O diagrama de caso do uso, o diagrama de atividade, o diagrama de seqüência e o diagrama da máquina do estado formam o conjunto de diagramas utilizados para representar o comportamento do sistema. O diagrama de caso de uso fornece uma descrição de alto nível das funcionalidades do sistema. O diagrama de atividade representa o fluxo dos dados e o controle existente entre as atividades realizadas pelo sistema. Um diagrama de seqüência representa a interação entre partes

colaborativas de um sistema. O diagrama de máquina de estado descreve as transições de estado e as ações que um sistema, ou suas partes, executam em resposta aos eventos.

O diagrama de requisito permite representar os requisitos, geralmente expressos em linguagem texto, através de elementos gráficos, além de permitir também relacionar estes requisitos a outros elementos do modelo do sistema. O diagrama de requisito representa as hierarquias e derivações existentes nos requisitos, permitindo também associar um determinado requisito a outro modelo que está tratando a sua satisfação e verificação. O diagrama de requisito provê uma ponte entre as ferramentas típicas de gerenciamento de requisitos e os modelos do sistema.

O diagrama de parâmetro representa as restrições impostas ao sistema tais como desempenho, confiabilidade e propriedades físicas. Atua como meio de integração das especificações e modelos de *design* com os modelos de análise da engenharia.

SysML também oferece um relacionamento de alocação que pode ser utilizado para alocar um conjunto de elementos de modelagem a outro, permitindo relacionar um comportamento à estrutura ou alocar lógica a componentes físicos. A alocação permite diversos tipos de relacionamento entre modelos de fases diferentes do desenvolvimento do projeto, como por exemplo: a ligação entre os requisitos e os elementos de *design*; o mapeamento do comportamento nas estruturas de implementação; e, também, a associação de partes do software com o desenvolvimento do hardware (OMGSysML, 2006).

2.5.3 Ferramentas de modelagem

Sendo SysML uma linguagem gráfica, a utilização de uma ferramenta CASE[12] proporciona o apoio adequado às atividades de modelagem, visualização e gerenciamento dos diversos diagramas oferecidos pela linguagem. Existem diversos fornecedores no mercado que oferecem ferramentas CASE com recursos para modelagem em UML 2 e que podem ser estendidas para suportar também a modelagem em SysML. O site oficial da OMG SysML[13] apresenta alguns destes fornecedores:

- ARTiSAN Software Tools: o Artisan Studio é a ferramenta para modelagem UML oferecida pela empresa e suporta a modelagem tanto em UML 2 como em SysML (ARTISAN, 2008).
- EmbeddedPlus Engineering: é uma empresa associada à IBM que oferece o EmbeddedPlus SysML Toolkit , uma extensão para as ferramentas de modelagem da IBM Rational[14] (EmbeddedPlus, 2008).
- No Magic: o MagicDraw é a ferramenta para modelagem UML oferecida pela empresa e suporta a modelagem tanto em UML 2 como em SysML (NoMagic, 2008).
- Papyrus: oferece a ferramenta de código aberto para modelagem gráfica em UML 2, o Papyrus 4 UML e uma extensão para tratamento SysML (Papyrus, 2008).

---

[12] Computer-aided software engineering

[13] OMG Systems Modeling Language, site oficial na internet: http://www.omgsysml.org/

[14] IBM Rational Software Modeler – RSM, IBM Rational Software Architect - RSA e IBM Rational Systems Developer - RSD (IBMRational, 2008).

2.5.4 Análise da SysML 1.0

SysML é uma linguagem para modelagem de um domínio específico, a Engenharia de Sistemas, e definida como uma personalização da UML 2.0 que, por sua vez, é uma linguagem de modelagem para uso geral. A vantagem em definir SysML como uma personalização da UML é a reutilização da notação e a da semântica da UML, ambas relativamente maduras. Entretanto, SysML também herda alguns problemas inerentes à UML, como a complexidade de notação e semântica e falhas no padrão de interoperabilidade entre diagrama, XMI (XML Metadata Interchange). (SysMLForum, 2007).

Uma das maiores evoluções encontradas em SysML, em relação à UML, é a possibilidade de representação de requisitos e de relacioná-los tanto com o modelo do sistema, como com a modelagem do *design* e com os procedimentos de teste (Vanderperren, 2005).

SysML é menor e mais fácil de aprender que UML, pois não possui as construções dedicadas ao tratamento de software e apresenta um número reduzido de tipos de diagrama. SysMl possui apenas nove diagramas em comparação aos treze existentes em UML (SysMLForum, 2007).

O racional, uma extensão de comentário utilizada em SysML, permite registrar as informações associadas a decisões em geral, principalmente decisões de *design*. Um racional pode ser ligado a qualquer entidade ou relacionamento, permitindo, por exemplo, associar elementos de *design* com requisitos, capturando e registrando o embasamento utilizado para uma determinada decisão de *design*, auxiliando a análise do impacto de uma alteração de requisitos no *design* do sistema (OMGSysML, 2006).

SysML introduz o diagrama de requisitos e define diversos tipos de relacionamentos para melhorar o rastreamento de requisitos. A finalidade não é substituir as ferramentas comerciais de gerência de requisitos, mas fornecer uma maneira padronizada para associar os requisitos ao *design* e teste. Os requisitos podem ser decompostos através da utilização do relacionamento de contenção, similar a forma utilizada em diagramas da classe. A dependência nomeada como rastreamento relaciona um determinado requisito derivado ao seu requisito origem. O *design* do sistema e os requisitos são ligados através da dependência satisfação. Finalmente, a dependência verificação associa um requisito com um caso de teste, utilizado para verificar estes requisitos. O diagrama de requisito contribui para a rigorosa transferência das especificações e informações relacionadas ao sistema entre as diversas ferramentas utilizadas pelos engenheiros de sistema, software e hardware (Vanderperren, 2005).

SysML introduz o conceito de montagem, um estereótipo de classe que descreve um sistema como uma estrutura de partes interconectada. Montagem fornece um elemento neutro de modelagem que pode ser utilizado para representar qualquer tipo de sistema, independentemente da natureza de seus componentes. Um modelo de montagem mostra a interconexão das partes de um sistema e suporta fluxos de informação entre os componentes (Vanderperren & Dehaene, 2005).

A introdução do conceito de alocação em SysML, provê um recurso genérico para relacionar um elemento de modelo a elementos de outro modelo. Além disso, causa o abandono do diagrama de comunicação e do diagrama de implantação existentes na UML. O conceito de alocação em SysML não está restrito apenas a alocação de artefatos de software aos componentes de hardware. Permite, também,

por exemplo, a ligação entre os requisitos e os elementos de *design* e o mapeamento do comportamento nas estruturas de implementação.

SysML introduz o conceito de controle de operação nos diagramas de atividades, o qual permite controlar a execução das atividades, habilitando ou desabilitando ações em função dos dados e parâmetros, sendo que em UML o controle permite apenas indicar quando uma ação se inicia (OMGSysML, 2006).

Parâmetros não fazem parte da UML e foram introduzidos como novo recurso em SysML, permitindo aos engenheiros de sistema representar e analisar as diversas restrições impostas ao sistema, tais como desempenho, confiabilidade e propriedades físicas. A análise é uma fase crítica na engenharia de sistemas, entretanto diversas linguagens que antecedem SysML, como UML e IDEF (IDEF, 2007), não resolvem de forma padronizada o problema de relacionar os requisitos e a especificação do sistema com os elementos de *design* produzidos na fase de análise. Os parâmetros de SysML permitem tratar este problema de forma padronizada, através de três elementos de modelagem: restrições, propriedades e parâmetros propriamente ditos. Restrições são análogas a fórmulas de equações. Um bloco de restrições é uma definição de formula que pode ser reutilizada e uma propriedade de restrição é um uso particular de um bloco genérico de restrição. Propriedades representam os valores associados a qualquer característica de um ambiente, sistema ou componente utilizado na análise do sistema. Parâmetros são as variáveis associadas a uma equação, ou restrição. Cabe observar que SysML não executa diretamente estas restrições, propriedades e parâmetros, os quais são passados para outras ferramentas de análise para serem computados (Peak, 2007a) (Peak, 2007b) (OMGSysML, 2006).

A SysML 1.0 não atende plenamente todos os requisitos da RFP. A matriz de conformidade dos requisitos (OMGSysML, 2006) possui um conjunto de requisitos obrigatórios parcialmente atendidos e alguns requisitos obrigatórios não atendidos, e que estão planejados para a versão 2.0. O mesmo ocorre com os requisitos opcionais.

M. Frapier e H. Habrias, no livro "*Software Specification Methods: An Overview Using a Case Study*", 2000, apresentam um processo para comparar, sem a intenção de classificação, métodos de especificação de software, (Frapier & Habrias, 00). Os atributos a serem avaliados na análise dos métodos de especificação estão apresentados na Tabela 3.

Tabela 3 - Atributos para análise dos métodos (Frapier & Habrias, 00)

#	Atributo	Valores
1	Paradigma	***state machine, algebra, process algebra, trace***
2	Formalismo	**Informal, semi-formal, formal**
3	Representação gráfica	**Sim, não**
4	Orientação objetos	**Sim, não**
5	Concorrência	**Sim, não**
6	Executabilidade	**Sim, não**
7	Uso de variáveis	**Sim, não**
8	Não determinismo	**Sim, não**
9	Lógica	**Sim, não**
10	Provabilidade	**Sim, não**
11	Verificação de modelo	**Sim, não**
12	Inibição de evento	**Sim, não**

A Tabela 4 apresenta a análise realizada pelo autor deste trabalho sobre a linguagem SysML 1.0 tomando como referência os atributos do processo de comparação de Frappier, onde se pode constatar que SysML atende a grande parte

dos atributos, tende a ser formal quando os requisitos da RFP forem atendidos e conta com um conjunto de profissionais potencialmente treinados na sua utilização pois é uma extensão de UML 2.

Tabela 4 - Análise da SysML 1.0

#	Atributo	SysML	Comentário
1	Paradigma	*state machine*	**Herança de UML através do diagrama de máquina de estado.**
2	Formalismo	Formal e semi-formal	**O conceito de alocação, a revisão dos diagramas herdados de UML e suas extensões favorecem as análises sintáticas, léxicas e semânticas livre de contexto. Entretanto os requisitos obrigatórios da RFP (OMGRFP, 03) não são totalmente atendidos na versão 1.0 da SysML**
3	Representação gráfica	Sim	**Herança da UML**
4	Orientação objetos	Sim	**Herança da UML**
5	Concorrência	Sim	**Herança da UML**
6	Executabilidade	Sim	**Os conceitos de controle de operação e de parâmetros oferecem recursos para simulação de alguns diagramas**
7	Uso de variáveis	Sim	**Parâmetros**
8	Não determinismo	Sim	**Parâmetros (fórmulas)**
9	Lógica	não	
10	Provabilidade	não	
11	Verificação de modelo	Sim	**Entretanto SysML 1.0 não atende plenamente a RFP**
12	**Inibição de evento**	**Sim**	**Viabilizado pela introdução dos conceitos de controle de operação e parâmetros.**

3 PROPOSTA DE MÉTODO PARA GESTÃO DE REQUISITOS

Este trabalho propõe um conjunto de atividades para a gestão de requisitos de sistemas, baseado em técnicas do escopo de engenharia de processos de negócio, de engenharia de requisitos, no processo unificado de desenvolvimento de software e na utilização de linguagens formais de modelagem, que seja aplicável pelos analistas de negócio e pelos engenheiros de requisitos e que agilize o processo de aceitação das especificações dos sistemas de informação por parte dos *stakeholders*, denominado neste estudo de Método para Gestão de Requisitos.

O Método para Gestão de Requisitos propõe processos que permitam a utilização da modelagem orientada a objetos e da linguagem UML versão 2, tanto na modelagem da visão da engenharia de requisitos, como na modelagem da visão da engenharia de processos de negócio, que serão executados, preferencialmente, por equipes independentes e possivelmente em diferentes janelas de tempo. Em ambas as visões, este trabalho propõe processos de gestão que viabilizem a transformação da modelagem em UML versão 2 para a linguagem formal de modelagem SysML. O Método para Gestão de Requisitos pretende obter uma visão unificada das modelagens formais dos requisitos e dos processos de negócio e que será utilizada, após as revisões e iterações necessárias, no tratamento da aceitação da especificação dos requisitos do sistema junto aos *stakeholders*.

A Figura 12 é uma representação gráfica do método proposto e auxilia um melhor entendimento dos processos que compõem o método, representando também a comunicação entre eles. Os retângulos com os cantos arredondados representam as etapas ou atividades de trabalho, os losangos representam tomadas de decisão, as setas direcionais cheias representam o fluxo básico das etapas de

trabalho e as setas direcionais tracejadas indicam a comunicação associada ao tratamento da gestão dos requisitos do sistema.
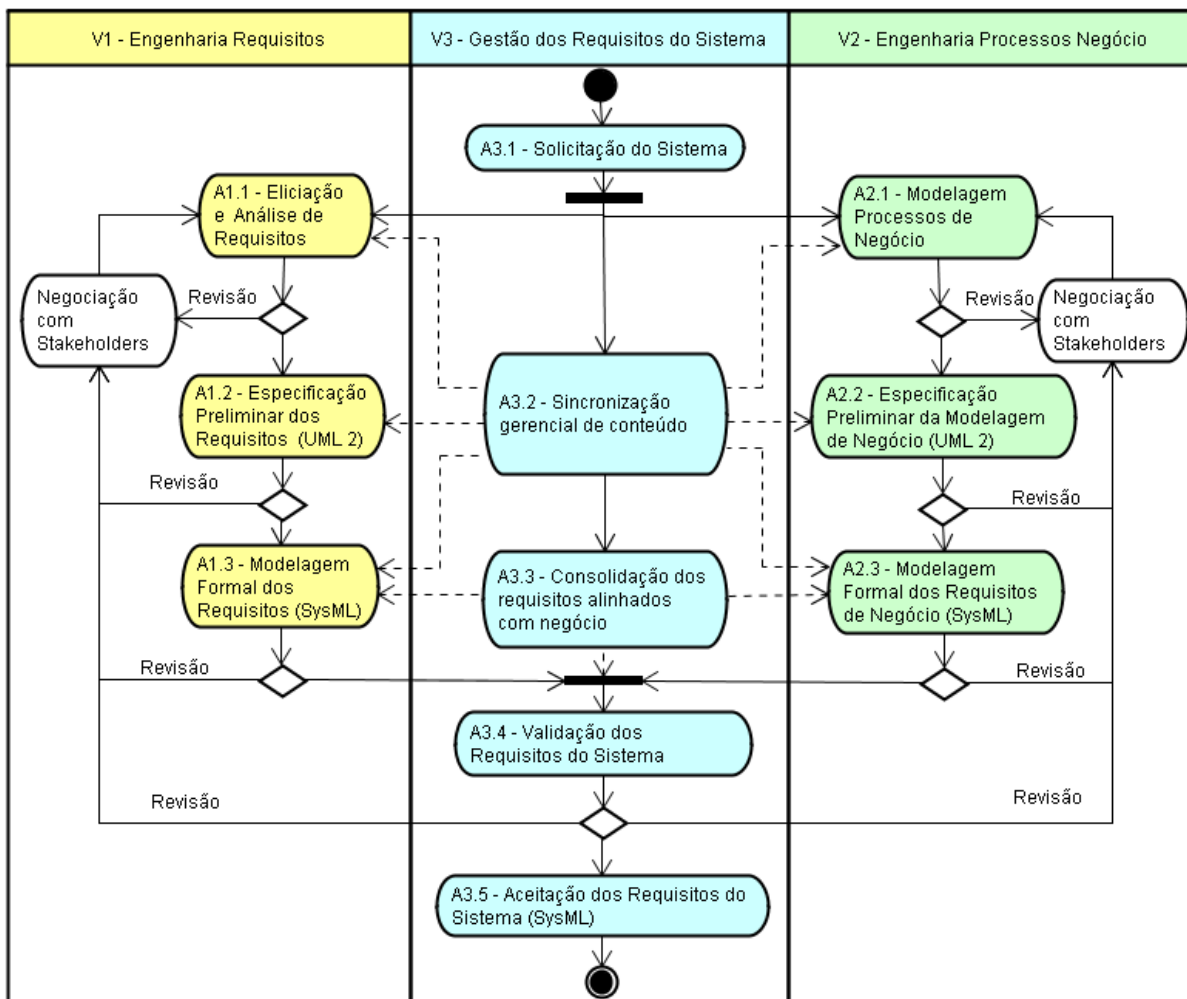
Figura 12 - Método proposto para gestão de requisitos

A coluna da esquerda apresenta as atividades associadas à Visão da Engenharia de Requisitos (V1), onde o objetivo é a eliciação e especificação dos requisitos do sistema, realizadas pela equipe de engenheiros de requisitos. A coluna da direita apresenta as atividades associadas à Visão da Engenharia de Processos de Negócio (V2), onde o objetivo é a modelagem de negócio associada ao sistema solicitado, realizadas pela equipe de analistas de negócio. A coluna central concentra as atividades de gestão e controle dos requisitos do sistema (V3), onde a

equipe de gestão do projeto atua desde a solicitação do sistema a ser desenvolvido até a fase final de aceitação dos requisitos do sistema junto aos *stakeholders*, direcionando a sincronização adequada das demais atividades que compõem o método de gestão de requisitos proposto.

3.1 VISÃO DA ENGENHARIA DE REQUISITOS (V1)

A Visão da Engenharia de Requisitos (V1) da Figura 12, neste estudo, é realizada através das seguintes atividades básicas:

- Eliciação e Análise dos Requisitos do Sistema (A1.1): atividade que recebe os dados associados ao contrato firmado na Solicitação do Sistema (A3.1) e, ao final do processo iterativo de negociação com os *stakeholders* fornece como saída um conjunto de requisitos documentados e priorizados do sistema que atendam às reais necessidades dos *stakeholders*.
- Especificação Preliminar dos Requisitos em UML 2 (A1.2): atividade que recebe os requisitos do sistema documentados e priorizados e, ao final do processo iterativo de negociação com os *stakeholders*, fornece como saída a modelagem dos requisitos do sistema representados na linguagem UML 2.
- Modelagem Formal dos Requisitos em SysML (A1.3): atividade que recebe a modelagem de requisitos do sistema em UML 2 como entrada e, ao final do processo iterativo de negociação com os *stakeholders*, fornece como saída a modelagem dos requisitos do sistema solicitado representado na linguagem formal SysML.

3.1.1 Eliciação e Análise dos Requisitos do Sistema (A1.1)

A eliciação de requisitos compreende o conjunto de atividades que a equipe formada por engenheiros de requisitos utiliza para levantar as solicitações dos *stakeholders*, buscando traduzir as reais necessidades nestas solicitações. As necessidades, expectativas e recursos esperados do sistema serão levantados junto a cada *stakeholder*. Também é necessário analisar os requisitos em relação a inconsistências e pontos incompletos, bem como negociar, solucionando conflitos, de forma que um conjunto de requisitos seja acordado. O objetivo é delimitar um conjunto de requisitos priorizados que atendam às reais necessidades dos *stakeholders* e estejam registrados no Documento de Requisitos.

A eliciação dos requisitos está associada ao processo de descoberta e análise do problema a ser resolvido, detectando o grau de importância de cada *stakeholder* e suas reais necessidades associadas ao problema, além de entender as restrições impostas ao sistema, devendo ser abordada por quatro dimensões: entendimento do domínio da aplicação, entendimento do problema, entendimento do negócio e entendimento das necessidades e restrições.

O processo de eliciação de requisitos apresenta quatro atividades básicas: ajuste dos objetivos da organização, aquisição de conhecimentos básicos do sistema, entendimento da organização e coleta de requisitos.

O Documento Requisitos é refinado através da execução das atividades do processo de análise de requisitos, representadas pelas atividades: verificação das necessidades, verificação de viabilidade e verificação de consistência e integridade.

Os resultados obtidos nesta atividade são registrados no Documento de Requisitos que contém a especificação de requisitos do sistema e é formado pelos artefatos:

- Modelo de casos de uso.
- Descrição dos casos de uso.
- Especificações suplementares.

A equipe de engenheiros de requisito deve realizar as atividades associadas à eliciação e a análise dos requisitos do sistema, sempre sob a supervisão da equipe de gestão para garantir a sincronização com os processos de negócio. Esta sincronização é obtida através de:

- Aplicação de padrões na utilização de nomes dos elementos de modelagem, com a utilização de vocabulário único e dicionário de nomes.
- Compartilhamento de dados e informações obtidas junto aos stakeholders.
- Otimização no acesso, entrevistas e reuniões com os *stakeholders*.

O resultado desta etapa do trabalho é apresentado pela equipe de engenheiros de requisitos aos *stakeholders* em reunião formal para validação e negociação do conteúdo, da qual participam também a equipe de gestão para tratamento de escopo e sincronização com o tratamento dos processos de negócio. Eventualmente esta atividade pode indicar a necessidade de revisão dos requisitos, desencadeando um processo iterativo onde as alterações são negociadas com os *stakeholders*.

3.1.2 Especificação Preliminar dos Requisitos em UML 2 (A1.2)

Da atividade de Eliciação e Análise dos Requisitos (A1.1) resulta o modelo de casos de uso e a especificação suplementar de requisitos do sistema, representados em parte de forma textual, oferecendo uma visão dos requisitos do sistema a ser desenvolvido.

Nesta etapa do trabalho, a equipe de trabalho formada por engenheiros de requisitos trabalha o Documento de Requisitos para que este esteja representado em UML 2, utilizando os diversos recursos e diagramas oferecidos pela linguagem, tais como: diagrama de casos de uso, diagrama de seqüência, diagrama de atividades e diagrama de máquina de estado. O objetivo é representar os requisitos no diagrama de casos de uso e nos diagramas de atividades associados a estes casos de uso. Os demais diagramas são utilizados pela equipe para auxiliar a eliciação dos requisitos e, muitas vezes, chegam a representar detalhes de *design* e implementação visando a esclarecer os requisitos e aumentar a compreensão das necessidades dos *stakeholders*.

Os requisitos não-funcionais definidos no documento de especificações suplementares devem ser refinados e detalhados e serão utilizados na etapa de Modelagem Formal dos Requisitos em SysML (A1.3).

O resultado desta etapa do trabalho é apresentado pela equipe de engenheiros de requisitos aos *stakeholders* em reunião formal para validação e negociação do conteúdo, da qual participam também a equipe de gestão para tratamento de escopo e sincronização com o tratamento dos processos de negócio. Eventualmente esta atividade pode indicar a necessidade de revisão dos requisitos, desencadeando um processo iterativo onde as alterações são negociadas com os *stakeholders*.

3.1.3 Modelagem Formal dos Requisitos em SysML (A1.3)

A atividade de Especificação Preliminar dos Requisitos em UML 2 (A1.2) resulta em diversos diagramas representando os requisitos e as necessidades dos *stakeholders* quanto ao sistema e, possivelmente, detalhes adicionais que auxiliaram a tarefa de eliciação dos requisitos.

SysML é uma extensão da UML versão 2.0, onde o diagrama de casos de uso é reutilizado da UML 2 e o diagrama de atividade de SysML é obtido através da extensão do diagrama de atividades da UML 2.0.

Nesta etapa do trabalho, a equipe formada por engenheiros de requisitos deve transferir o diagrama de casos de uso e os diagramas de atividades associados aos casos de uso representados em UML 2 para SysML, os demais diagramas utilizados em UML 2 foram utilizados apenas para auxiliar a eliciação dos requisitos.

Utilizando a ferramenta de modelagem UML 2.0, os engenheiros de requisitos exportam os diagramas para arquivos XML, os quais são importados na ferramenta de modelagem SysML. Os diagramas de casos de uso importados podem ser tratados diretamente na ferramenta de modelagem SysML. Os diagramas de atividades importados podem ser retrabalhados pela equipe de trabalho para incorporar as extensões de SysML que permitem controlar a execução das atividades, habilitando ou desabilitando ações em função de dados e parâmetros, sendo que em UML o controle permite apenas indicar quando uma ação se inicia.

A equipe de trabalho formada por engenheiros de requisitos utiliza o diagrama de requisitos de SysML para representar os requisitos não-funcionais descritos no documento de especificações suplementares.

O resultado desta etapa do trabalho é apresentado pela equipe de engenheiros de requisitos aos *stakeholders* em reunião formal para validação e negociação do conteúdo, da qual participam também a equipe de gestão para tratamento de escopo e sincronização com o tratamento dos processos de negócio. Eventualmente, esta atividade pode indicar a necessidade de revisão dos requisitos, desencadeando um processo iterativo onde as alterações são negociadas com os *stakeholders*.

3.2 VISÃO DA ENGENHARIA DE PROCESSOS DE NEGÓCIO (V2)

A Visão da Engenharia de Processos de Negócio (V2) da Figura 12, neste estudo, é realizada através das seguintes atividades básicas:

- Modelagem dos Processos de Negócio (A2.1): atividade que recebe os dados associados ao contrato firmado na Solicitação do Sistema (A3.1) e, ao final do processo iterativo de negociação com os *stakeholders*, fornece como saída uma visão ampla da modelagem dos processos de negócio da empresa, tendo como foco principal os processos de negócio associados ao sistema solicitado.
- Especificação Preliminar da Modelagem de Negócio em UML 2 (A2.2): atividade que recebe a modelagem dos processos de negócio da empresa e, ao final do processo iterativo de negociação com os *stakeholders*, fornece como saída a modelagem dos processos de negócio da empresa em UML 2, com ênfase para os requisitos de negócio associados ao sistema solicitado.
- Modelagem Formal dos Requisitos de Negócio em SysML (A2.3): atividade que recebe a modelagem de negócio em UML 2 como entrada e, ao final do processo iterativo de negociação com os *stakeholders*, fornece como saída a modelagem dos requisitos de negócio associados ao sistema solicitado representado na linguagem formal SysML.

3.2.1 Modelagem dos Processos de Negócio (A2.1)

Nesta etapa a equipe de trabalho formada por analistas de negócio deve levantar e documentar os objetivos, processos e recursos do negócio, considerando as oportunidades de melhoria associadas aos processos de negócio atuais, para

que estes possam ser remodelados numa situação futura e para que, a partir deles, possam ser derivados os casos de uso de negócio. Sempre sob a supervisão da equipe de gestão para garantir a sincronização dos processos de negócio com o foco do trabalho que é o tratamento dos requisitos do sistema a ser desenvolvido, a equipe de analistas de negócio deve realizar os seguintes passos:

1. Capturar um vocabulário de negócio comum que possa ser utilizado em todas as descrições textuais do negócio.
2. Identificar os objetivos da organização como um todo e, através de desdobramentos hierárquicos, obter sub-objetivos mais detalhados, permitindo a rastreabilidade dos objetivos estratégicos até os objetivos operacionais.
3. Identificar e modelar os processos de negócio atuais, descrevendo a seqüência de atividades que devem ser executadas em cada processo, para se alcançar os objetivos definidos anteriormente. Além disso, o analista de negócio também deve identificar e modelar os recursos (sistemas, unidades organizacionais responsáveis e informações de entrada e saída) que serão necessários para a execução dos processos de negócio.
4. Associar os objetivos identificados aos processos de negócio que irão suportá-los, onde os relacionamentos entre objetivos e processos podem ser: um para um, um para muitos e muitos para muitos.
5. Identificar as necessidades de melhorias nos processos de negócio, identificando e priorizando os principais problemas.
6. Modelar os novos processos e, caso necessário, os novos recursos que serão implantados para a execução do negócio.

7. Identificar os casos de uso de negócio associados aos processos de negócio. A derivação dos casos de uso dos processos de negócio pode ser uma relação um para muitos. Desta forma, um processo ou uma atividade de um processo, dependendo do nível de detalhamento em que estejam modelados, pode gerar apenas um ou vários casos de uso.

O resultado obtido na modelagem dos processos de negócio é o diagrama de casos de uso de negócio, onde a descrição do fluxo de trabalho relacionado a um processo de negócio pode ocorrer de duas formas:

- Representação textual com a utilização de um formato básico de descrição, descrevendo o fluxo normal e os fluxos alternativos.
- Utilização do diagrama de atividades, descrevendo graficamente os passos do caso de uso de negócio.

O nível de detalhe adequado para identificação e descrição dos casos de uso de negócio, bem como a padronização da nomenclatura a ser utilizada nos elementos que compõem a modelagem são definidos pelo analista de negócio e pela equipe de gestão do projeto.

O resultado desta etapa do trabalho é apresentado pela equipe de analista de negócio aos *stakeholders*, em reunião formal para validação e negociação do conteúdo, da qual participam também a equipe de gestão para tratamento de escopo e sincronização com o tratamento dos requisitos do sistema a ser desenvolvido. Eventualmente, esta atividade pode indicar a necessidade de revisão dos processos de negócio, desencadeando um processo iterativo onde as alterações são negociadas com os *stakeholders*.

3.2.2 Especificação Preliminar Modelagem de Negócio UML 2 (A2.2)

A atividade de Modelagem dos Processos de Negócio (A2.1) resultou o diagrama de casos de uso de negócio, formado pelos casos de uso de negócio e a descrição dos fluxos de trabalho de forma textual ou utilizando o diagrama de atividades, oferecendo uma visão dos processos de negócio onde o sistema a ser desenvolvido está contido.

O diagrama de atividades, dentre os diagramas da UML 2, é o que mais se aproxima dos modelos tradicionais de modelagem de processos de negócio. Dessa forma, esse tipo de diagrama, inicialmente concebido com o intuito de modelar o fluxo de trabalho relacionado ao comportamento de um sistema, também pode ser utilizado para modelagem de processos de negócio, podendo ser denominado de diagrama de atividades de negócio.

Assim como os demais diagramas tradicionais utilizados para modelagem de processos de negócio, esse diagrama permite modelar o fluxo de trabalho considerando situações de sincronismo e de decisão; permite relacionar, através de colunas, papéis e unidades organizacionais responsáveis pela execução das atividades do processo; e também pode mostrar, através de extensões da UML 2, os comportamentos e relacionamentos entre as entidades de negócio e as atividades em execução.

Nesta etapa do trabalho, a equipe formada por analistas de negócio deve priorizar os casos de uso de negócio, tendo como foco os requisitos do sistema a ser desenvolvido, eliminar toda representação textual utilizada para a descrição dos fluxos de trabalho nos casos de uso de negócio e obter um modelo de casos de uso de negócio representado em UML 2. Sempre sob a supervisão da equipe de gestão para garantir a sincronização dos processos de negócio com o foco do trabalho que

é o tratamento dos requisitos do sistema a ser desenvolvido, a equipe de analistas de negócio deve realizar os seguintes passos:

1. Priorizar os casos de uso de negócio que oferecem requisitos para o sistema a ser desenvolvido, com a orientação da equipe de gestão e mantendo o sincronismo com o tratamento de requisitos do sistema.
2. Retrabalhar os casos de uso de negócio, com representação textual dos fluxos de trabalho, para que estes sejam representados no diagrama de atividades da UML 2.
3. Elaborar o diagrama de casos de uso de negócio em UML 2, tendo como foco o tratamento de requisitos do sistema.

A priorização dos casos de uso de negócio é orientada pela equipe de gestão de projetos, mantendo a sincronização dos processos de negócio com o foco do trabalho que é o tratamento dos requisitos do sistema a ser desenvolvido.

O resultado desta etapa do trabalho é apresentado pela equipe de analista de negócio aos *stakeholders* em reunião formal para validação e negociação do conteúdo, da qual participam também a equipe de gestão para tratamento de escopo e sincronização com o tratamento dos requisitos do sistema a ser desenvolvido. Eventualmente esta atividade pode indicar a necessidade de revisão dos processos de negócio, desencadeando um processo iterativo onde as alterações são negociadas com os *stakeholders*.

3.2.3 Modelagem Formal dos Requisitos de Negócio em SysML (A2.3)

A atividade de especificação preliminar da modelagem dos Processos de Negócio em UML 2 (A2.2) resulta o diagrama de casos de uso de negócio e os

diagramas de atividade associados aos fluxos de trabalho, com ênfase para os requisitos de negócio associados ao sistema a ser desenvolvido.

SysML é uma extensão da UML versão 2.0, onde o diagrama de casos de uso é reutilizado da UML 2 e o diagrama de atividade de SysML é obtido através da extensão do diagrama de atividades da UML 2.0.

Nesta etapa do trabalho, a equipe de trabalho, formada por analistas de negócio, deve transferir o diagrama de casos de uso de negócio e os diagramas de atividades em UML 2 para SysML.

Utilizando a ferramenta de modelagem UML 2.0, os analistas de negócio exportam os diagramas para arquivos XML e estes são importados na ferramenta de modelagem SysML. Os diagramas de casos de uso importados podem ser tratados diretamente na ferramenta de modelagem SysML. Os diagramas de atividades importados podem ser retrabalhados pelos analistas de negócio para incorporarem as extensões de SysML, que permitem controlar a execução das atividades, habilitando ou desabilitando ações em função de dados e parâmetros. Em UML o controle permite apenas indicar quando uma ação inicia.

Ainda nesta etapa, os analistas de negócio, auxiliados por engenheiros de requisitos da equipe de gestão do projeto, tomando como referência diagrama de casos de uso de negócio e os diagramas de atividades, realizam uma estimativa dos requisitos para o sistema a ser desenvolvido, através da criação de um modelo de casos de uso do sistema. A realização do modelo de casos de usos do sistema é feita através dos seguintes passos:

- Verificar se um ator de negócio utilizará o sistema e, em caso afirmativo, criar um ator para o sistema no modelo de casos de uso do sistema com o mesmo nome do ator de negócio.

- Para cada caso de uso de negócios em que o ator de negócio, que foi incluído no modelo de casos de uso do sistema, criar um caso de uso de sistema.
- Repetir estes passos para todos os atores de negócio.

O resultado desta etapa do trabalho é apresentado pela equipe de analista de negócio aos *stakeholders*, em reunião formal para validação e negociação do conteúdo, da qual participam também a equipe de gestão para tratamento de escopo e sincronização com o tratamento dos requisitos do sistema a ser desenvolvido. Eventualmente esta atividade pode indicar a necessidade de revisão dos processos de negócio, desencadeando um processo iterativo onde as alterações são negociadas com os *stakeholders*.

3.3 GESTÃO DOS REQUISITOS DO SISTEMA (V3)

Este estudo propõe que a Gestão dos Requisitos do Sistema (V3) da Figura 12, seja realizada através das seguintes atividades básicas:

- Solicitação do Sistema (A3.1): atividade que representa a iniciação do projeto, estabelecendo o contrato firmado junto aos *stakeholders* para o desenvolvimento do sistema e definição do escopo e das restrições iniciais do sistema.
- Sincronização Gerencial de Conteúdo (A3.2): atividade realizada pela equipe de gestão do projeto atuando na formação e capacitação adequada das equipes, na comunicação entre as equipes de engenharia de requisitos e processos de negócio, e na utilização adequada dos insumos e artefatos parciais obtidos por estas equipes.

- Consolidação dos Requisitos Alinhados com Negócio (A3.3): atividade realizada por representantes das equipes de engenharia de requisitos e processos de negócio, onde ocorre a sincronização entre o modelo em SysML obtido na visão da engenharia de requisitos e o modelo em SysML obtido na visão da engenharia de processos de negócio.
- Validação dos Requisitos do Sistema (A3.4): atividade que recebe a consolidação dos requisitos alinhados com o negócio modelados em SysML e que através de um processo iterativo de negociação, o qual tem a participação de *stakeholders*, analistas de negócio, engenheiros de requisitos e equipe de *design*, fornece como saída a modelagem formal dos requisitos do sistema representada em SysML, alinhada com os processos de negócio da empresa e validada pelos *stakeholders*.
- Aceitação dos Requisitos do Sistema em SysML (A3.5): atividade que recebe a modelagem formal dos requisitos do sistema em SysML, fornecendo para a negociação de aceitação dos requisitos junto aos *stakeholders* a possibilidade de rastrear cada um dos requisitos dentro dos processos de engenharia de requisitos e de engenharia de processos de negócio. Esta atividade fornece como saída a modelagem formal dos requisitos do sistema, definindo o escopo do projeto de desenvolvimento e oferecendo insumos adequados para as estimativas de tamanho e esforço.

3.3.1 Solicitação do Sistema (A3.1)

Atividade que representa a iniciação do projeto, estabelecendo o contrato firmado junto aos *stakeholders* para o desenvolvimento do sistema e definindo o escopo e as restrições iniciais do sistema.

Nesta etapa do trabalho são realizadas as seguintes atividades:

- Preparar o ambiente para o projeto, avaliando o projeto e a organização, selecionando ferramentas e decidindo quais partes do processo devem ser melhoradas.
- Obter e documentar uma descrição inicial do sistema a ser desenvolvido. A descrição do sistema terá menos detalhes nas fases iniciais do projeto e mais detalhes nas fases finais, uma vez que as características do sistema são progressivamente elaboradas nas demais etapas.
- Atingir o consenso entre a equipe de gestão de projetos e os *stakeholders* sobre os objetivos do ciclo de vida do projeto, indicando os resultados obtidos a cada etapa e o grau de comprometimento esperado dos *stakeholders*.
- Obter e organizar a documentação já existente na organização e que será utilizada como insumo para as etapas seguintes.
- Tratar o planejamento de alocação das equipes de trabalho, tanto para o tratamento da visão da engenharia de processos de negócio formada por analistas de negocio, como para o tratamento da visão da engenharia de requisitos, formada por engenheiros de requisitos.
- Treinar a equipe de trabalho para a correta utilização dos métodos, técnicas e ferramentas a serem utilizados.
- Analisar a possível defasagem de tempo entre o tratamento da visão da engenharia de processos de negócio e o tratamento da visão da engenharia de requisitos, permitindo planejar a sincronização dos trabalhos e a consolidação dos resultados.

Esta etapa fornece o material de entrada e estrutura os recursos necessários para a realização das atividades associadas à Visão da Engenharia de Requisitos (V1) e à Visão da Engenharia de Processos de Negócio (V3), além de preparar a equipe de trabalho e recursos necessários para as demais atividades da gestão de requisitos do sistema.

3.3.2 Sincronização Gerencial de Conteúdo (A3.2)

A atividade de Solicitação do Sistema (A3.1) fornece o material de entrada, os direcionadores de trabalho e o planejamento de recursos necessários para a realização das atividades associadas à visão da engenharia de requisitos e à visão da engenharia de processos de negócio.

Nesta atividade é proposto que a equipe de gestão atue na manutenção da comunicação e sincronização dos resultados obtidos pelas duas visões, realizando as seguintes atividades:

- Manter o ambiente de trabalho através da manutenção das ferramentas de trabalho, incluindo diagramadores UML e SysML, esquemas de controle de versão e padronização de documentos.
- Manter o dicionário de dados e vocabulário.
- Manter a alocação das equipes de trabalho, tanto para o tratamento da visão da engenharia de processos de negócio formada por analistas de negocio, como para o tratamento da visão da engenharia de requisitos, formada por engenheiros de requisitos.
- Oferecer treinamentos complementares aos integrantes das equipes das duas visões, garantindo a homogeneidade dos trabalhos e padronização dos resultados.

- Tratar a possível defasagem de tempo entre o tratamento da visão da engenharia de processos de negócio e o tratamento da visão da engenharia de requisitos, permitindo a sincronização dos trabalhos e a consolidação dos resultados.
- Tratar e gerenciar a gestão de mudanças.
- Tratar a gestão da comunicação com os *stakeholders*, nas duas visões e atuar como facilitador nesta comunicação, evitando retrabalhos e acessos desnecessários e duplicados aos *stakeholders*.

Nesta atividade a equipe de gestão atua gerencialmente tanto nas atividades Eliciação e Análise de Requisitos (A1.1), Especificação Preliminar dos Requisitos em UML 2 (A1.2) e Modelagem Formal dos Requisitos em SysML (A1.3), associadas à Visão da Engenharia de Requisitos (V1), como nas atividades Modelagem dos Processos de Negócio (A2.1), Especificação Preliminar da Modelagem de Negócio em UML 2 (A2.2) e Modelagem Formal dos Requisitos de Negócio em SysML (A2.3), associadas à Visão da Engenharia de Processos de Negócio, garantindo que os resultados das atividades (A1.3) e (A2.3) sejam entregues adequadamente para a atividade Consolidação dos Requisitos Alinhados com o Negócio (A3.3).

A eventual defasagem de tempo entre o tratamento da Visão da Engenharia de Requisitos (V1) e a Visão da Engenharia de Processos de Negócio (V2), onde as atividades A2.1 e A2.2 são realizadas antecipadamente às atividades A1.1 e A1.2, ou até mesmo não são realizadas resultando na inexistência de uma modelagem de negócio, a sincronização Gerencial do conteúdo deve atuar alocando analistas de negócio para obter uma modelagem de negócio que adequada para a realização da atividade Modelagem Formal dos Requisitos de Negócio em SysML (A2.3).

Esta etapa garante, através de uma atuação gerencial, que as atividades Modelagem Formal dos Requisitos em SysML (A1.3) e Modelagem Formal dos Requisitos de Negócio em SysML (A2.3) sejam realizadas de acordo com as definições propostas nesta pesquisa, fornecendo os insumos necessários para a realização da atividade de consolidação dos requisitos alinhados com negócio.

3.3.3 Consolidação dos Requisitos Alinhados com Negócio (A3.3)

Esta atividade deverá ser coordenada pela equipe de gestão e recebe como entrada os modelos em SysML obtidos pela Modelagem Formal dos Requisitos do Sistema (A1.3) e Modelagem Formal dos Requisitos de Negócio (A3.3), resultados das atividades associadas, respectivamente, à Visão da Engenharia de Requisitos (V1) e à Visão da Engenharia de Processos de Negócio (V3), e fornece um modelo único em SysML e consolidado, onde os requisitos do sistema estão em concordância com os requisitos de negócio.

O trabalho realizado na atividade de Sincronização Gerencial de Conteúdo (A3.2), busca garantir que os dois modelos representados em SysML estejam construídos sob as mesmas regras de padronização de nomes, vocabulário, estrutura de diagramação dos modelos e relações de dependência entre elementos dos modelos.

As ferramentas para diagramação SysML oferecem recursos para a representação da estrutura dos modelos indicando as dependências entre os elementos que compõem a linguagem e, também, fornecendo recursos para validação sintática da construção dos diagramas e das dependências entre eles. Estas ferramentas oferecem também recursos baseados na construção sintática dos modelos listando as diferenças entre dois modelos, que possuem a mesma estrutura

de construção, e orientam na detecção de inconsistência e falhas nas relações de dependência entre os elementos.

A facilidade de verificação de diferenças sintáticas entre modelos oferece um instrumento para a equipe de gestão, auxiliada por analistas de negócio e engenheiro de requisitos, detectarem as inconsistências entre os dois modelos e realimentar tanto a atividade de Modelagem Formal dos Requisitos do Sistema (A1.3), como a atividade de Modelagem Formal dos Requisitos de Negócio (A2.3), para que os modelos evoluam e ofereçam o grau de consistência adequado. Permitindo, assim, que seja obtido um modelo único contemplando as duas visões e contando com as relações de dependências corretas e consistentes, bem como oferecendo, desta forma, um caminho de rastreabilidade entre os requisitos do sistema e os requisitos de negócio atendidos pelo sistema e que estão alinhados com o modelo de negócio da organização.

As ferramentas para diagramação SysML oferecem também recursos para a fusão de dois modelos, em que os elementos com mesmo nome e sintaxe são fundidos e relatórios para tratamento de diferenças são oferecidos permitindo que eventuais inconsistências entre os modelos possam ser tratadas. Este recurso de fusão de modelos permite selecionar dois modelos diferentes e gerar um terceiro, representando a fusão dos dois primeiros modelos, obtendo-se, desta forma, um modelo de requisitos único, contemplando as necessidades de negócio e que deve facilitar o processo de validação e aceite junto aos *stakeholders*.

3.3.4 Validação dos Requisitos do Sistema em SysML (A3.4)

Esta atividade tem como objetivo estabelecer que os requisitos estejam definidos de forma correta, consistente e integra. O processo de validação dos

requisitos envolve os *stakeholders*, a equipe de gestão, analistas de negócio e engenheiros de requisitos. Tem como objetivo a análise do documento de requisitos, buscando problemas, omissões e ambigüidades. Nesta atividade pode haver também o envolvimento das equipes de *design* e implementação do projeto, com o objetivo de detectar eventuais dificuldades de implementação ou melhorias decorrentes de soluções tecnológicas, fornecendo insumos para aprimoramentos e correções nos requisitos.

O principal problema na validação dos requisitos é que não há uma referência documentada para ser utilizada como base na validação do documento de requisitos. A validação de requisitos é um processo longo, envolvendo um grupo heterogêneo de pessoas que estão tratando um extenso e complexo documento.

A atividade de Consolidação dos Requisitos Alinhados Negócio (A3.3) resulta em um documento de requisitos representado por um modelo em SysML e obtido através da consolidação dos requisitos do sistema frente aos processos de negócio.

Nesta etapa, o documento de requisitos será apresentado pela equipe de trabalho aos *stakeholders*, buscando a validação dos requisitos, utilizando uma documentação gráfica, de fácil compreensão e contendo elementos de rastreabilidade com os requisitos e processos de negócio da organização. Obtém-se, assim, um instrumento eficiente para sustentar um processo convergente e rápido de convencimento dos *stakeholders* quanto à completude e à cobertura dos requisitos do sistema a ser desenvolvido.

O trabalho será coordenado pela equipe de gestão com a participação de engenheiros de requisitos e analistas de negócio. O resultado do trabalho será uma versão refinada do documento de requisitos e será apresentado aos *stakeholders* em reunião formal para validação e negociação do conteúdo. Eventualmente esta

atividade poderá indicar a necessidade de revisão dos requisitos ou até mesmo da modelagem de negócio, desencadeando um processo iterativo onde as alterações são negociadas com os *stakeholders*.

3.3.5 Aceitação dos Requisitos do Sistema em SysML (A3.5)

A atividade Validação dos Requisitos do Sistema em SysML (A3.4) fornece o documento de requisitos onde os requisitos do sistema estão representados em SysML e compatíveis com os processos de negócio.

A equipe de gestão, juntamente com os engenheiros de requisitos, calculará as estimativas de tamanho, esforço e prazo para o desenvolvimento do sistema.

O documento de requisitos, junto com as estimativas de tamanho, esforço e prazo, será apresentado aos *stakeholders*, com o objetivo de obter o aceite e a aprovação de seu conteúdo.

O resultado desta atividade será o documento de requisitos, junto com as estimativas de tamanho, esforço e prazo, aceito e aprovado pelos *stakeholders*, oferecendo às equipes de desenvolvimento um instrumento representado em uma linguagem formal, provavelmente reduzindo o volume de interpretações semânticas.

4 CASOS DE APLICAÇÃO DO MÉTODO PROPOSTO

Este capítulo tem como objetivo ilustrar o método proposto para gestão de requisitos através da apresentação de dois casos, um associado ao desenvolvimento de sistema de informação e outro ao desenvolvimento de um sistema de automação de máquinas de venda.

A utilização de ferramenta CASE para a modelagem dos processos de negócio e dos requisitos oferece recursos para modelagem, visualização e gerenciamento dos diversos diagramas que compõem o documento de requisitos. Existem diversos fornecedores que oferecem ferramentas para modelagem UML 2 e que podem ser estendidas para suportar a modelagem em SysML. O site oficial da OMG SysML[15] apresenta diversos fornecedores:

- ARTiSAN Software Tools: o Artisan Studio é a ferramenta para modelagem UML oferecida pela empresa e suporta a modelagem tanto em UML 2 como em SysML (ARTISAN, 2008).
- EmbeddedPlus Engineering: é uma empresa associada a IBM que oferece o EmbeddedPlus SysML Toolkit, uma extensão para as ferramentas de modelagem da IBM Rational[16] (EmbeddedPlus, 2008).
- No Magic: o MagicDraw é a ferramenta para modelagem UML oferecida pela empresa e suporta a modelagem tanto em UML 2 como em SysML (NoMagic, 2008).

---

[15] OMG Systems Modeling Language, site oficial na internet: http://www.omgsysml.org/

[16] IBM Rational Software Modeler – RSM, IBM Rational Software Architect - RSA e IBM Rational Systems Developer - RSD (IBMRational, 2008).

- Papyrus: oferece a ferramenta de código aberto para modelagem gráfica em UML 2, o Papyrus 4 UML, e uma extensão para tratamento SysML (Papyrus, 2008).

A modelagem dos dois casos foi realizada na ferramenta MagicDraw, contando com licenças de utilização de software temporária fornecida pelo fabricante especificamente para o desenvolvimento deste estudo.

Cabe ressaltar que a descrição dos casos foi inserida neste trabalho com o intuito de melhor explicitar o método proposto e não pretende demonstrar os artefatos gerados, apresentando apenas aqueles mais relevantes.

4.1 CASO DESENVOLVIMENTO DE SISTEMA DE INFORMAÇÃO

Este caso associado ao desenvolvimento de sistema de informação foi baseado na tese de doutorado apresentada no programa de pós-graduação da Faculdade de Engenharia Mecânica da UNICAMP (Kintschner, 2003) que teve como base um estudo em empresas do setor têxtil.

O objetivo de apresentar este caso é discutir, principalmente, as atividades de Modelagem Formal dos Requisitos em SysML (A1.3), Modelagem Formal dos Requisitos de Negócio (A2.3) e Consolidação dos Requisitos Alinhados com Negócio (A3.3) propostas no Modelo de Gestão de Requisitos da Figura 12.

Visando a um melhor entendimento do caso de aplicação será apresentada descrição do modelo de negócio básico associado ao processo têxtil.

4.1.1 Descrição do Processo têxtil

O processo têxtil consiste de dois processos básicos de negócio, a fiação e a tecelagem (Corder, 1994). A Figura 13 apresenta o modelo de casos de uso de negócio do processo têxtil.
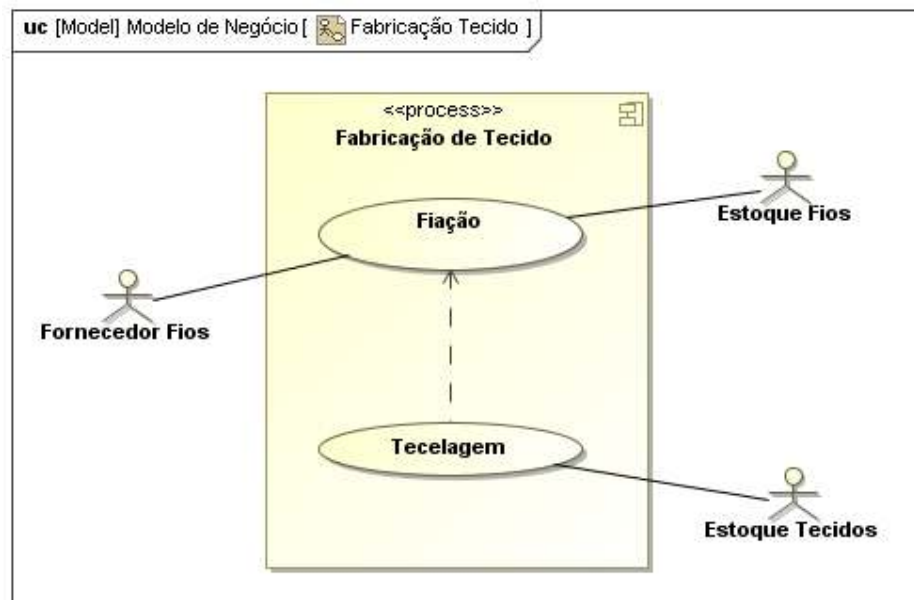
Figura 13 - Processo Têxtil: Casos de Uso Negócio

O processo de fiação é a etapa inicial do processo produtivo têxtil, que se inicia com a obtenção das fibras geradoras, que podem ser do tipo natural, artificial ou sintética.

A fiação consiste na paralelização das fibras e na aderência de uma às outras por atrito e no torcimento para que adquiram resistência, resultando em fios homogêneos preparados para a fabricação de tecidos. O processo de fiação para os diversos tipos de fibras geradoras é muito parecido.

A fase de fiação resulta em fios penteados ou cardados, sendo que a qualidade e a regularidade diferenciam os dois tipos. O fio penteado oferece uma melhor qualidade devido ao maior número de processos de limpeza a que é submetido,

diminuindo a quantidade de resíduos nas fibras, e uma melhor regularidade, em razão da redução nas torções sofridas, resultando em uma textura mais lisa e refinada. Este tipo de fio é indicado para a produção de tecidos destinados à confecção, como camisas e lençóis. O fio cardado oferece uma menor qualidade, devido ao maior número de torções, e é utilizado para a produção de jeans, tecidos para estofados e cortinas entre outros.

A Figura 14 apresenta o diagrama de atividades do processo de fiação do fio cardado.
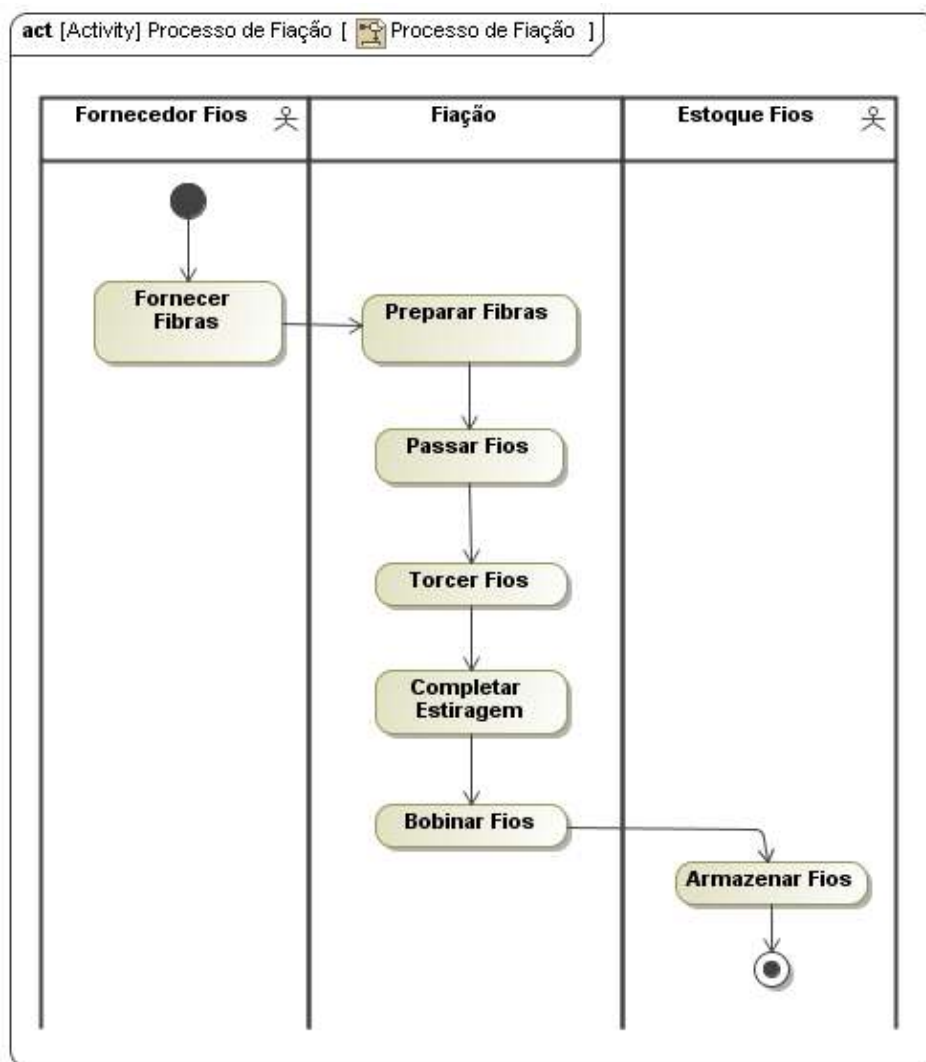
Figura 14 - Processo de Fiação: Diagrama de Atividades

O processo de tecelagem consiste em dispor os fios que estão enrolados nas bobineiras e subdividi-los em urdume e trama. No urdume, os fios são reunidos em paralelo e determinam o comprimento do tecido, pois estão dispostos longitudinalmente nos teares e permitem a execução da trama e produzir o tecido. A Figura 15 apresenta o diagrama de atividades do processo de tecelagem.
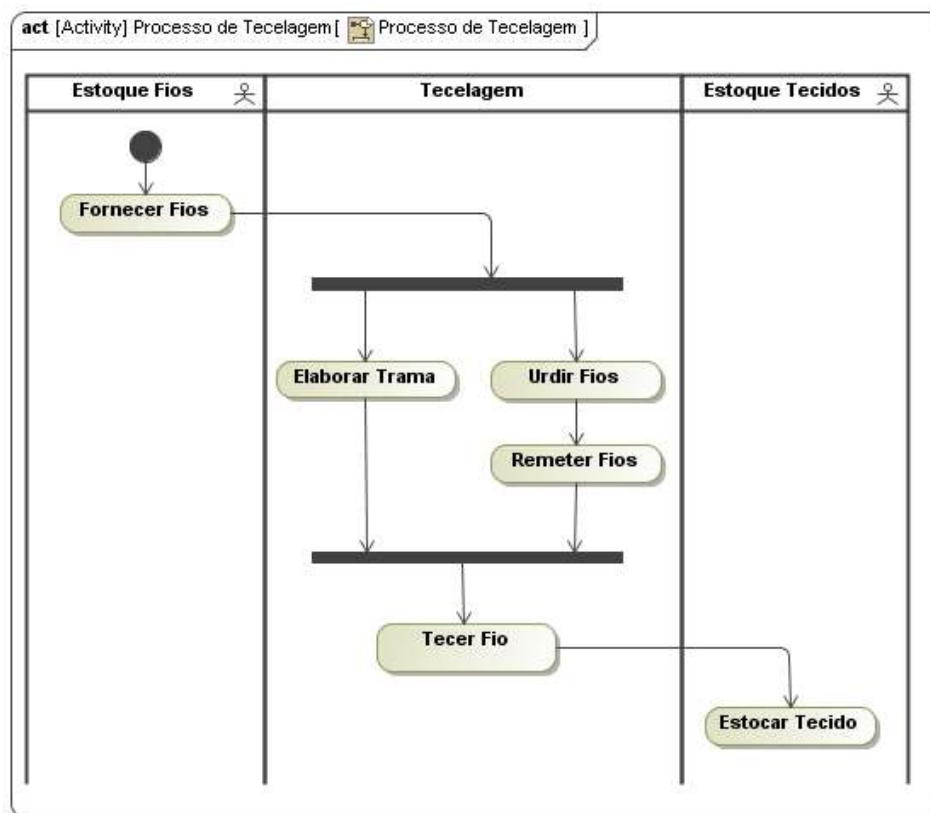
Figura 15 - Processo de Tecelagem: Diagrama de Atividades

A atividade Solicitação do Sistema (A3.1) da Figura 12 representa a iniciação do projeto, estabelecendo o contrato firmado junto aos *stakeholders* para o desenvolvimento do sistema e definindo o escopo e as restrições iniciais do sistema, que, neste caso, é o desenvolvimento de um sistema de informação capaz de suportar e controlar o processo produtivo associado à linha de produção do tecido jacquard, utilizado geralmente para decoração.

4.1.2 Visão da Engenharia de Processos de Negócio (V2)

As atividades Modelagem Processos de Negócio (A2.1) e Especificação Preliminar da Modelagem de Negócio em UML 2 (A2.2) da Figura 12, neste caso, foram suprimidas e o resultado para a atividade Modelagem Formal dos Requisitos de Negócio em SysML (A2.3) está representada pelo diagrama de casos de usos de negócio e pelos diagramas de atividades associados aos casos de uso.

A Figura 16 apresenta o diagrama de casos de uso de negócio associados à produção de tecido jacquard, representando a relação entre os atores e os casos de usos de negócio.
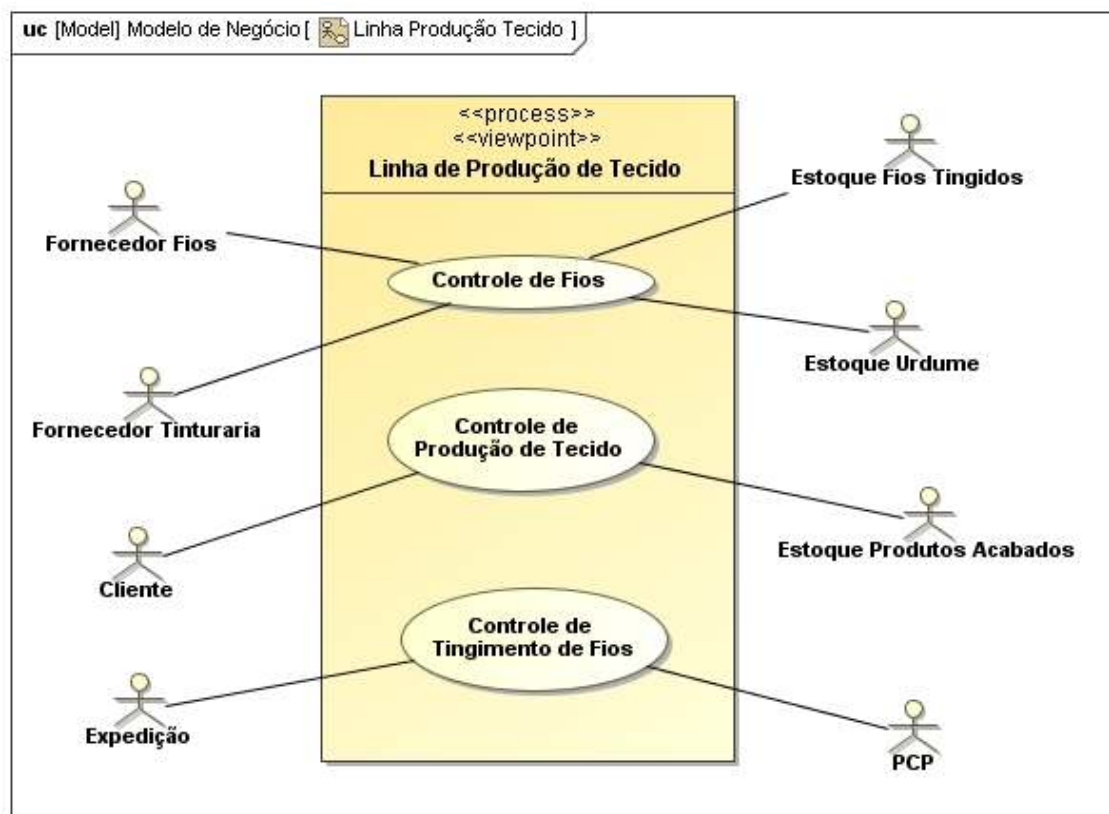
Figura 16 - Linha de Produção: Casos de Uso de Negócio

O caso de uso de negócio controle de fios está detalhado no diagrama de atividades da Figura 17, que representa o processo de controle do estoque de fios

crus e tingidos e o gerenciamento da entrada e saída de fios para o tingimento em empresa terceirizada.
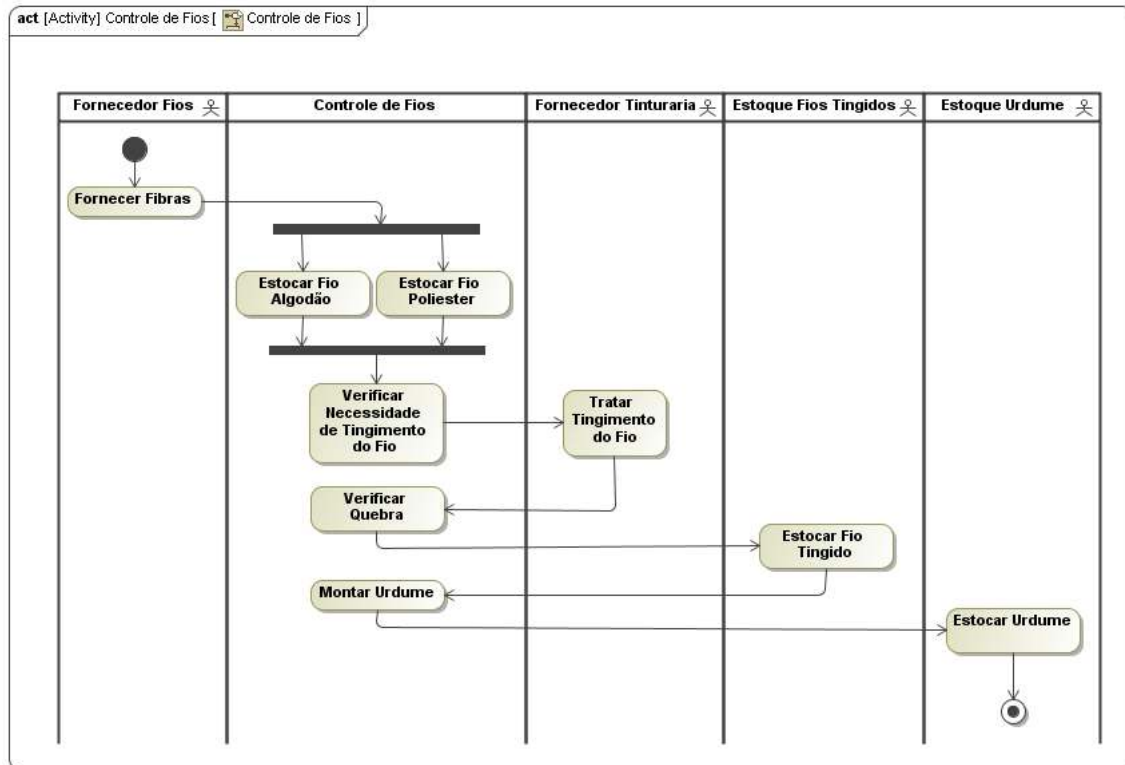
Figura 17 - Controle de Fios: Diagrama de Atividades

Segue a descrição das atividades:

- Estocar fio algodão: as caixas de fio de algodão cru são recebidas e armazenadas em local reservado para esse tipo de fio.
- Estocar fio de poliéster: as caixas de fio de poliéster são recebidas e armazenadas em local reservado para esse tipo de fio.
- Verificar necessidade de tingimento de fio: a partir de verificação do ponto mínimo no estoque de fios de algodão e poliéster e os novos pedidos a serem produzidos, são definidos os rolos de fios que serão enviados para a empresa terceirizada para serem tingidos.

- Verificar quebra: para cada retorno de tingimento de fios em empresa terceirizada é verificada a diferença existente entre a quantidade em quilos enviada e recebida.
- Estocar fio tingido: o fio tingido recebido é estocado de tal forma que facilite sua manipulação para envio à produção.
- Montar urdume: a partir dos pedidos de venda são montados os rolos urdumes que serão utilizados para a fabricação dos tecidos.

O caso de uso controle de produção de tecido está detalhado no diagrama de atividades da Figura 18.
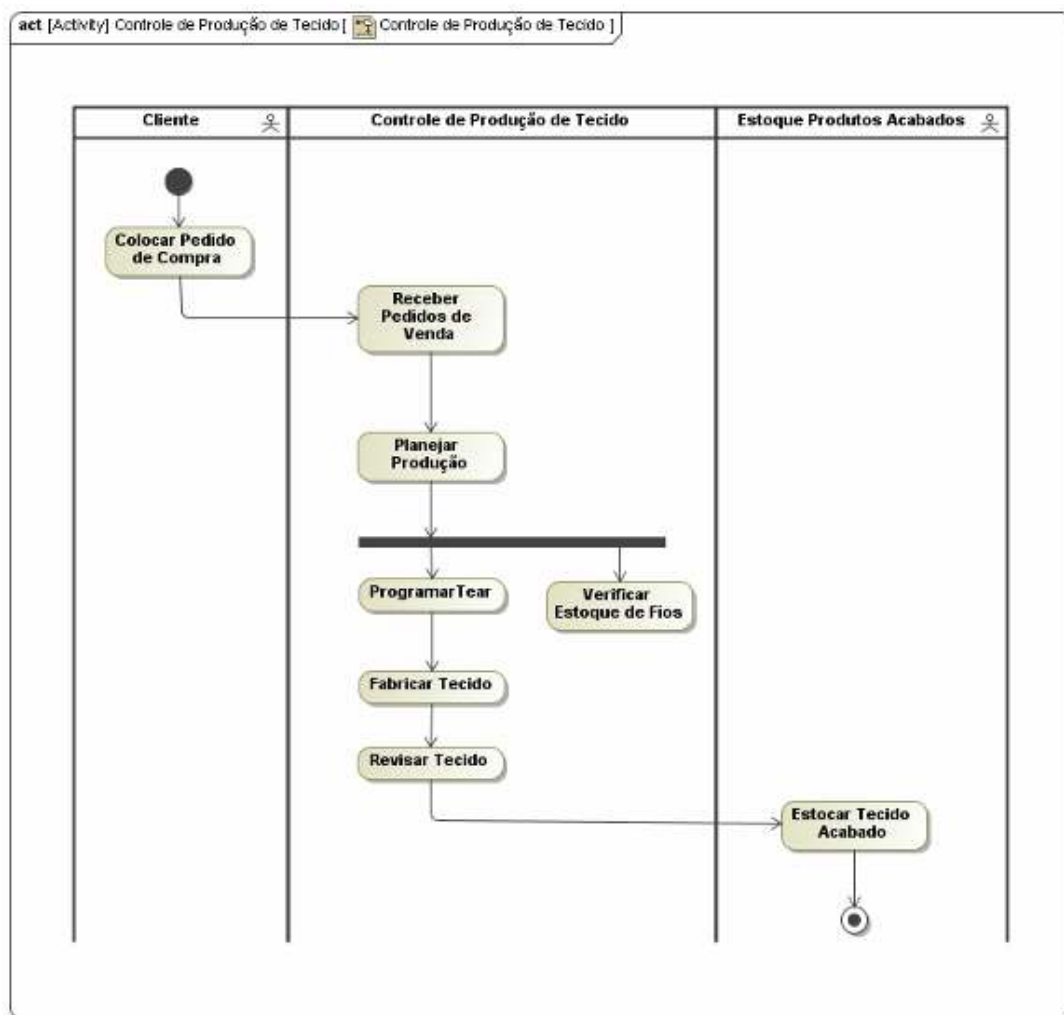
Figura 18 - Controle de Produção de Tecido: Diagrama de Atividades

Segue a descrição das atividades:

- Receber pedidos de venda: os pedidos de venda são recebidos via fax ou e-mail e são agrupados semanalmente para definição de programação de produção.
- Planejar produção: a partir dos pedidos recebidos na semana é definida a programação de produção de acordo com os artigos, desenhos, cor de urdume e cor da trama.
- Verificar estoque de fios: a partir do planejamento da produção é verificada a quantidade de fios tingidos no estoque e a necessidade de envio de fios para tingimento.
- Programar tear: com a definição da programação da produção e das cores dos urdumes a serem utilizados, é definida a programação dos teares disponíveis.
- Fabricar tecido. o urdume é engrupado e a trama é montada no tear, permitindo a fabricação do tecido a partir destes dois fios.
- Revisar tecido. o tecido produzido é revisado para verificar se houve algum defeito na produção.
- Estocar tecido acabado. o rolo de tecido acabado é revisado e é estocado até sua expedição.

O caso de uso de negócio controle de tingimento de fios está detalhado no diagrama de atividades da Figura 19, representando o processo de tingimento, que é realizado em empresa terceirizada.
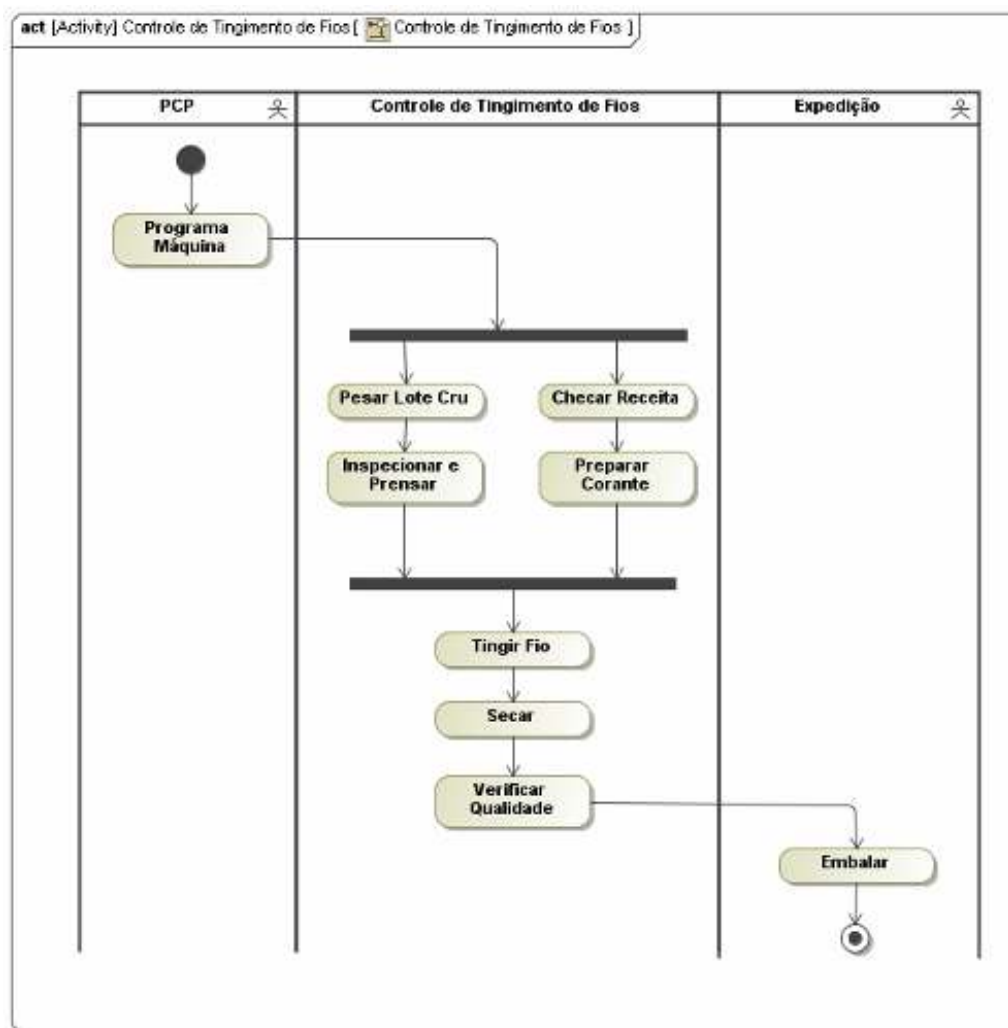
Figura 19 - Controle de Tingimento de Fios: Diagrama de Atividades

Segue a descrição das atividades:

- Programar máquina: com o recebimento da listagem de lotes a serem tingidos, é realizada a conferência da lista, com a verificação da prioridade da produção e da existência da receita de tingimento. Quando do início da preparação do processo de tingimento, é iniciado o preenchimento do formulário de acompanhamento do processo de tinturaria (FAPT), que será completado no decorrer do processo.

- Pesar lote cru: as rocas de fios que irão ser tingidos são pesadas. São preenchidas no FAPT as seguintes informações: total de quilos do lote e o nome do funcionário que realizou a pesagem.
- Inspecionar e prensar: os fios são colocados em cestos e inspecionados visualmente e em seguida são prensados. São preenchidos no FAPT: número da prensa, número do cesto e nome do funcionário que realizou o trabalho.
- Checar receita: este procedimento é realizado quando não há receita relacionada com a programação do tingimento ou quando há mais de dois meses esta cor não é utilizada.
- Preparar corante: são separados e pesados os corantes que serão utilizados e, em seguida, preparada a receita.
- Tingir fio: os cestos são transportados para a máquina de tingimento e, em seguida, são colocados os corantes. Durante o tingimento, o funcionário faz verificações relativas ao PH e à pressão. O processo de tingimento leva em média de duas a três horas. São preenchidos no FAPT: número da máquina, PH tingimento, PH redução, PH amaciante, pressão, data do tingimento e o nome do funcionário que executou o tingimento.
- Secar: após o tingimento, os cestos são levados para a máquina de secagem. São preenchidos no FAPT: hora início da secagem, hora fim da secagem e o funcionário que realizou a secagem.
- Verificar qualidade: após a secagem, são retiradas e analisadas amostras de três rolos do lote. Caso estejam dentro do padrão, o lote é

liberado para embalagem. Caso seja detectado algum problema, são retiradas mais amostras e enviadas para o laboratório.

- Embalar: após a liberação, as rocas são embaladas e enviadas para a expedição.

Tomando como referência o diagrama de casos de uso de negócio e os diagramas de atividades, a Figura 20 apresenta uma estimativa dos requisitos para o sistema de informação obtidos da Visão da Engenharia de Negócio (V2).

Figura 20 - Estimativa para os requisitos do sistema

4.1.3 Visão da Engenharia de Requisitos (V1)

As atividades Eliciação e Análise de Requisitos (A1.1) e Especificação Preliminar dos Requisitos em UML 2 (A1.2) da Figura 12, neste caso, foram suprimidas e o resultado para a atividade Modelagem Formal dos Requisitos em SysML (A1.3) está representado pelo diagrama de casos de uso e pelo diagrama de requisitos.

A Figura 21 apresenta o diagrama de casos de uso associados ao sistema de informação para suporte e controle da linha de produção do tecido jacquard.
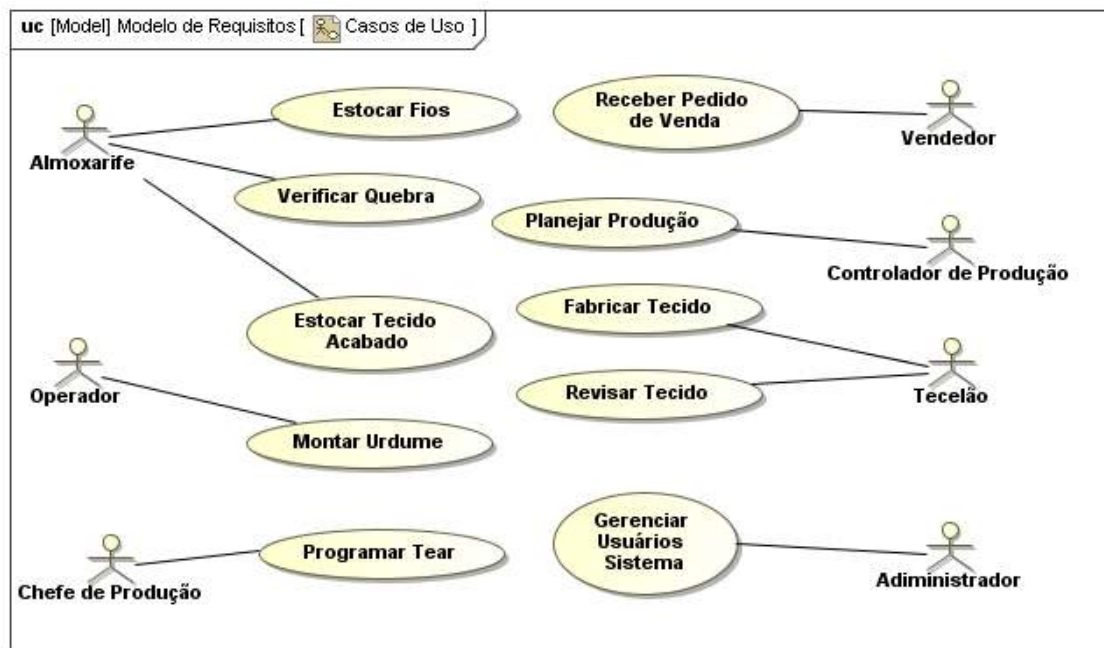
Figura 21 - Requisitos funcionais: Diagrama de Casos de Uso

Segue a descrição de cada caso de uso do sistema:

- Receber pedido de venda: trata do cadastro do pedido de venda do cliente informando: CNPJ, nome e endereço (rua, cidade, estado, telefone, fax, e-mail) do cliente, artigo (desenho, cor urdume, cor trama), quantidade, data do pedido e data provável para entrega.

- Estocar fios: trata do recebimento das caixas de fios no almoxarifado e para cada recebimento é informado o fornecedor, lote do fornecedor, data de produção, número da nota fiscal, peso bruto, peso líquido e quantidade de rocas.
- Verificar quebra: trata da verificação de quebra e consiste em gerenciar a quantidade de fios recebidos que apresentam quebras. Deve ser informado: tipo de fio, data do envio, saldo do fio em estoque, saldo acumulado enviado, quantidade de saída, número da nota fiscal de referência, data do recebimento, cor, peso da entrada, peso da quebra, número da nota fiscal referência.
- Estocar tecido acabado: o sistema sugere local de armazenagem do tecido (almoxarifado e quadra) até a expedição do mesmo, sendo permitido ao almoxarife trocar o local definido.
- Planejar produção: os pedidos de venda são agrupados semanalmente por desenho, cor de urdume e cor da trama. O resultado desta atividade é um relatório informando a quantidade de peças que deverão ser produzidas na semana.
- Programar tear: a partir do planejamento da produção é definido o que deverá ser produzido em cada tear, através do agrupamento por cor do urdume, pois esse é o item com maior dificuldade de ser alterado em um tear. A ficha de programação do tear gerada tem as seguintes informações: número do tear, nome do artigo, data, cor, quantidade (peças) e observação.

- Montar urdume: trata do preenchimento do formulário de fabricação do rolo de urdume com as seguintes informações: número do rolo de urdume, números dos lotes dos fios, quantidade de fios, peso, metros do rolo e data de fabricação do rolo.
- Fabricar tecido: com a fabricação do tecido a partir do rolo de urdume e das rocas de trama são gravadas as seguintes informações sobre o rolo de tecido: cliente, data de produção, lote de rocas utilizadas de trama, lote de rolo de urdume utilizado, metragem produzida, peso da peça produzida.
- Revisar tecido: trata da verificação da qualidade do tecido, que só será liberado se estiver de acordo com o padrão. Caso o tecido não seja liberado será informado o problema ocorrido e o tecido fica bloqueado até a resolução do problema.
- Gerenciar Usuários Sistema: trata do cadastro e controle de acesso dos usuários do sistema.

A Figura 22 apresenta o diagrama de requisitos onde os requisitos não funcionais estão modelados.
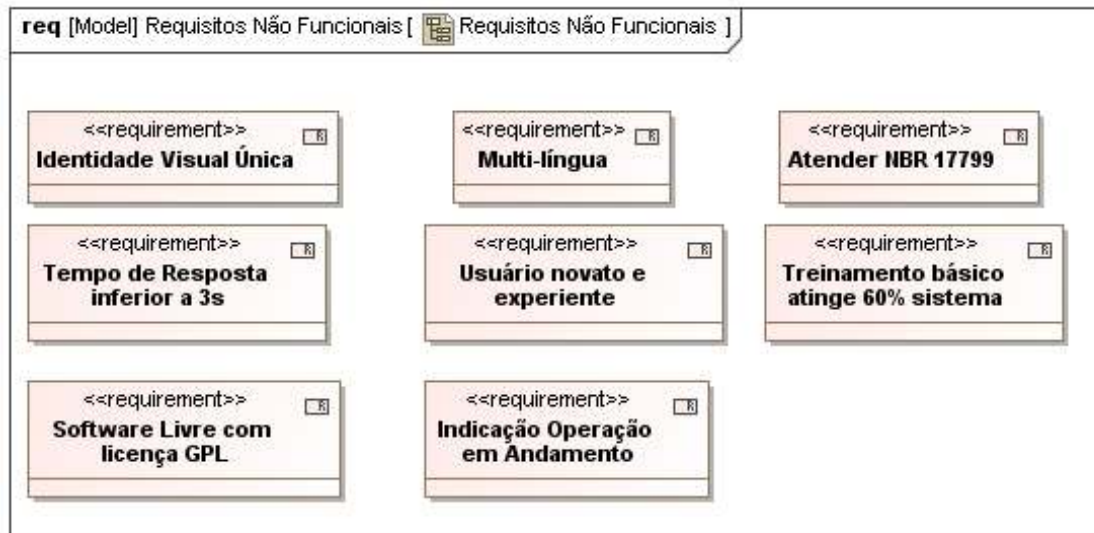
Figura 22 - Requisitos não funcionais: Diagrama de Requisitos

4.1.4 Visão da Gestão de Requisitos do Sistema (V3)

A atividade de Solicitação do Sistema (A3.1) da Figura 12 estabeleceu o contrato com os *stakeholders* para o desenvolvimento do sistema de informação para suporte e controle da linha de produção do tecido jacquard.

Na atividade Sincronização Gerencial de Conteúdo (A3.2) foram mantidas as ferramentas de trabalho, o controle de versões, a padronização de documentos e o dicionário de dados e vocabulário utilizados na Visão da Engenharia de Requisitos (V1) e na Visão da Engenharia de Processos de Negócio (V2).

Na atividade Consolidação dos Requisitos Alinhados com Negócio (A3.3) os resultados obtidos na Visão da Engenharia de Requisitos (V1) e na Visão da Engenharia de Processos de Negócio (V3) são comparados com o recurso de comparação de projetos oferecido pela ferramenta de modelagem, cujo resultado está apresentado na Figura 23.
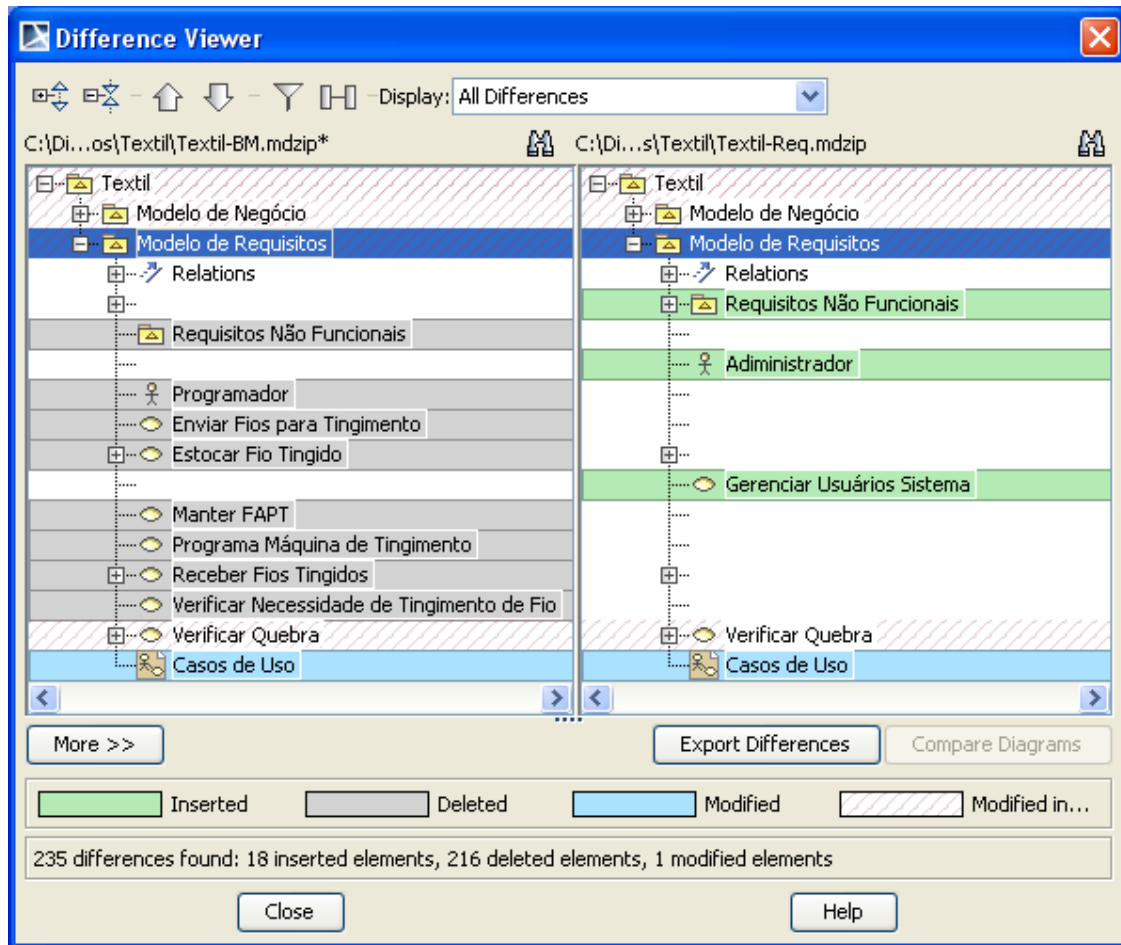
Figura 23 - Diferenças nas visões de requisitos e negócios

O lado esquerdo da Figura 23 apresenta os elementos que compõem a Visão da Engenharia de Processos de Negócios (V2) e o lado direito os elementos que compõem a Visão da Engenharia de Requisitos (V1).

Considerando-se a legenda da Figura 23, onde a cor cinza indica elementos que foram excluídos, observa-se que o resultado da comparação apresenta um conjunto de casos de uso e um ator do sistema que estão sendo considerados na Visão da Engenharia de Processos de Negócios (V2) e que não estão na Visão da Engenharia de Requisitos (V1). Analisando-se estes casos de uso observa-se que o caso de uso de negócio Controle de Tingimento de Fios não foi considerado na V1 e que deve ser incluído para o desenvolvimento do sistema.

Por outro lado, a legenda verde na Figura 23 indica que elementos foram inseridos na V1, um caso de uso e um ator, ambos associados ao tratamento do controle de acesso ao sistema, e que não foram considerados na V2.

O resultado final da atividade Consolidação dos Requisitos Alinhados com Negócio (A3.3) é a união dos modelos de V1 e V2, que é obtida com o recurso de fusão de projetos oferecido pela ferramenta de modelagem.

A Figura 24 apresenta a tela intermediária de fusão das visões V1 e V2. Necessário observar que as ações a serem executada sobre os elementos estão indicadas na legenda e que existe um tratamento para inibir ou efetivar as ações. A marcação em vermelho inibe a ação e a marcação em verde efetiva a ação indicada na legenda.
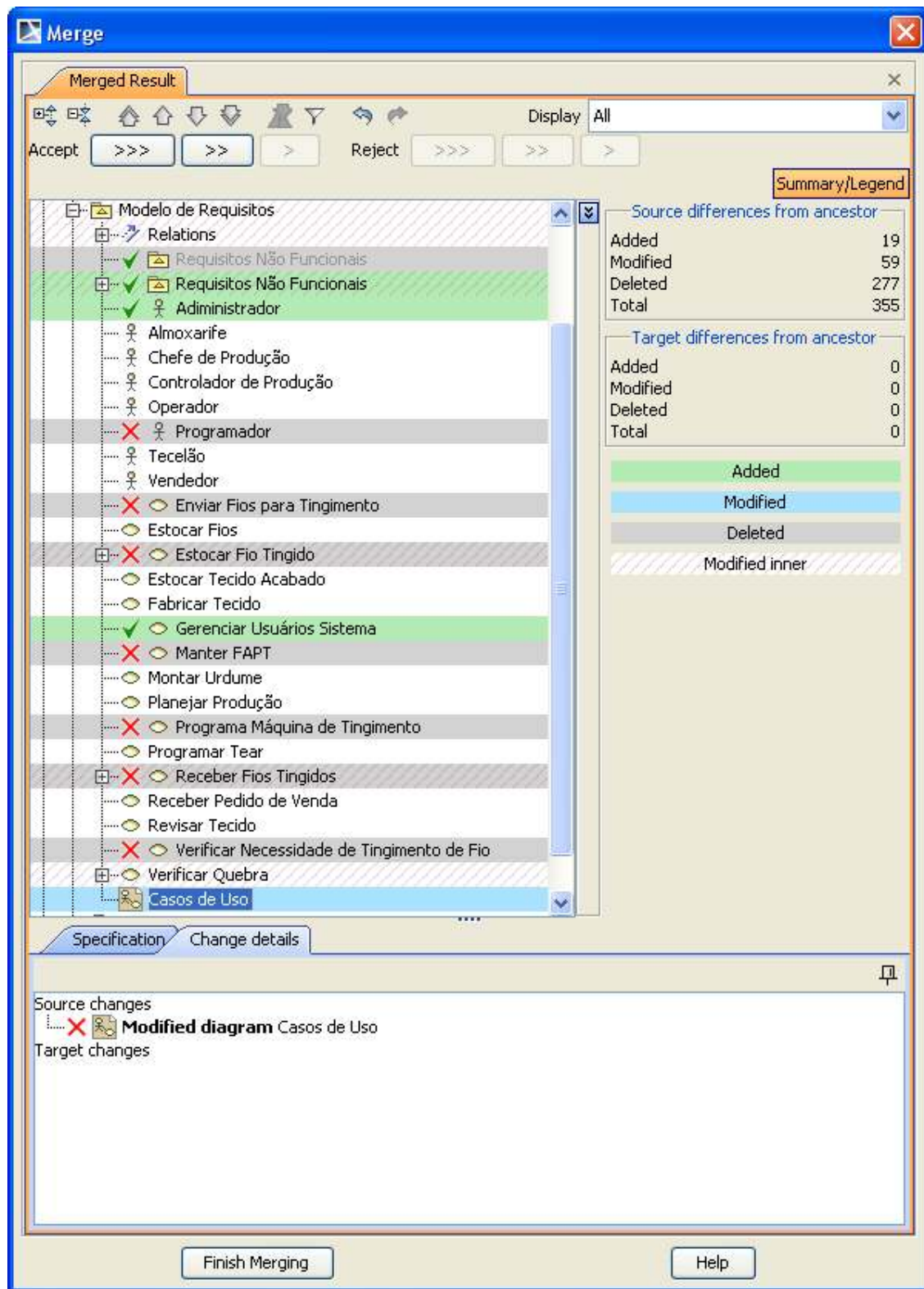
Figura 24 - Tela de fusão dos projetos

A finalização da fusão das visões V1 e V2 resulta em um modelo contendo a modelagem de requisitos do sistema contemplando as necessidades de negócio. A

Figura 25 apresenta o modelo de casos de uso do sistema de informação para suporte e controle da linha de produção do tecido jacquard.
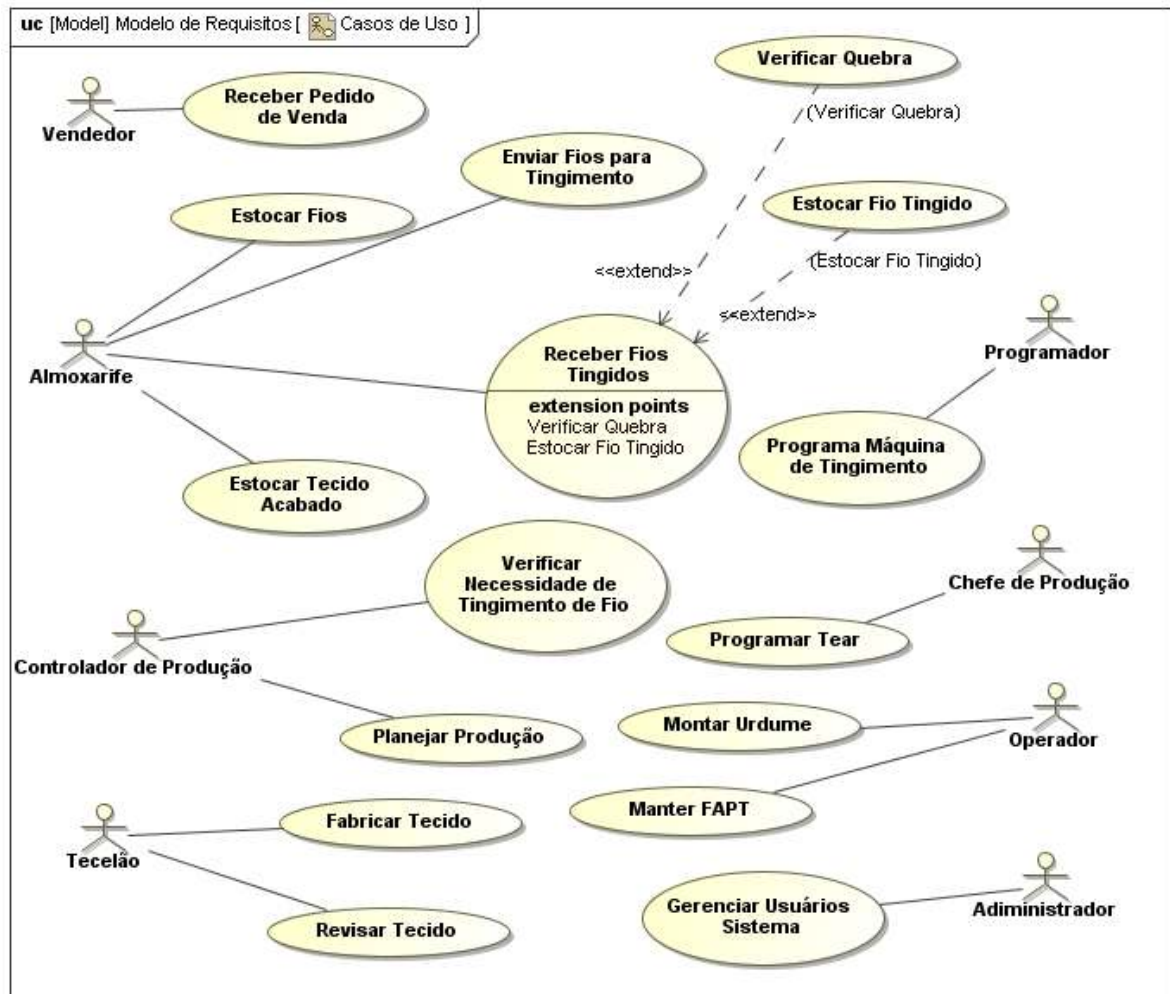
Figura 25 - Modelo de Casos de Uso do Sistema

As atividades de Validação dos Requisitos do Sistema (A3.4) e Aceitação dos Requisitos do Sistema (A3.5) são realizadas tomando como referência o modelo de casos de usos da Figura 25.

4.2 DESCRIÇÃO DO CASO DE AUTOMAÇÃO

Este caso associado ao desenvolvimento de sistema de automação foi baseado na experiência obtida em um projeto real de uma empresa. Este projeto ainda está em andamento e as informações apresentadas neste estudo representam os resultados obtidos até o momento.

O objetivo de apresentar este caso é apresentar, de forma parcial, a aplicação do método de gestão de requisitos na especificação de novos produtos. Estes produtos fazem parte da evolução tecnológica pretendida pela empresa para melhor atender seus clientes. Neste caso será tratado especificamente um produto associado ao controle e supervisão de máquina de venda, pretendido e encomendado por um cliente específico.

4.2.1 Descrição da Empresa

A Signalcard Tecnologia ltda é uma empresa que desenvolve e comercializa soluções para automação de vendas, controle de acesso, tarifação de serviços e transporte público. Grande parte destas soluções está baseada na utilização de cartão indutivo, tecnologia desenvolvida pela Signalcard e suportada por diversas patentes.

No cartão indutivo, os dados estão armazenados em células. Cada célula possui um pequeno fusível feito de uma liga metálica. Como uma memória de computador, é possível associar códigos a cada célula do cartão, de modo que o valor é "1" se a célula estiver contínua e "0" se o fusível foi fundido e, conseqüentemente, a célula estiver interrompida (Figura 26). Normalmente, todas as células do cartão são produzidas com valor "1" e são gravadas nas unidades de

leitura e gravação de cartão. Esta gravação é única e não retornável, garantindo a alta segurança do cartão indutivo. Além disso, o cartão indutivo não pode ser produzido, copiado ou duplicado fora da indústria, o que lhe confere características anti-fraude com alta relação custo-benefício.
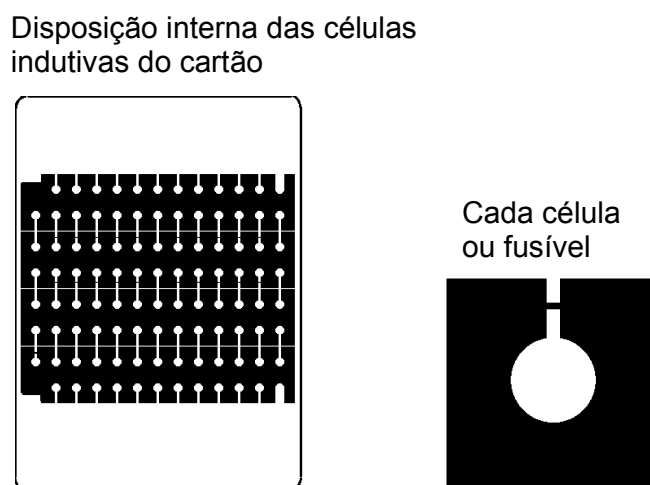
Figura 26 - Estrutura interna do cartão indutivo

A evolução dos meios e das redes de comunicação oferece oportunidades para o desenvolvimento de uma nova família de produtos e serviços, principalmente na área de supervisão e controle de máquinas de venda (VM[17]). Este caso de aplicação terá como foco o desenvolvimento de sistemas de automação associados à evolução da linha de produtos de VM.

4.2.2 Sistema de Supervisão de Máquinas de Venda

A Figura 27 apresenta a estrutura proposta para o sistema para supervisão de máquina de vendas, o qual é composto pelos seguintes elementos:

---

[17] Do termo em inglês *vending machines*

- Supervisor Central: os dados das VMs são recebidos pela internet, tratados e armazenados. Permite emitir relatórios em tempo real do status de funcionamento de cada VM e relatórios gerenciais de consumo, falhas e faturamento.
- Área Fechada: compreende uma região fechada com uma rede local (LAN) instalada e com acesso à Internet (hotéis, parques, empresas, universidades, indústrias, etc). Possui diversas VM distribuídas e interligadas através de rede wireless (WLAN) ou ethernet.
- VM – Conexão Mobile: VM em local com acesso à rede de telefonia celular e conectada à Internet através de um modem GSM/GPRS/EDGE.
- Grupo de VM – Conexão Mobile: conjunto de VMs interligadas por uma rede local ethernet e conectadas à Internet através de um modem GSM/GPRS/EDGE.
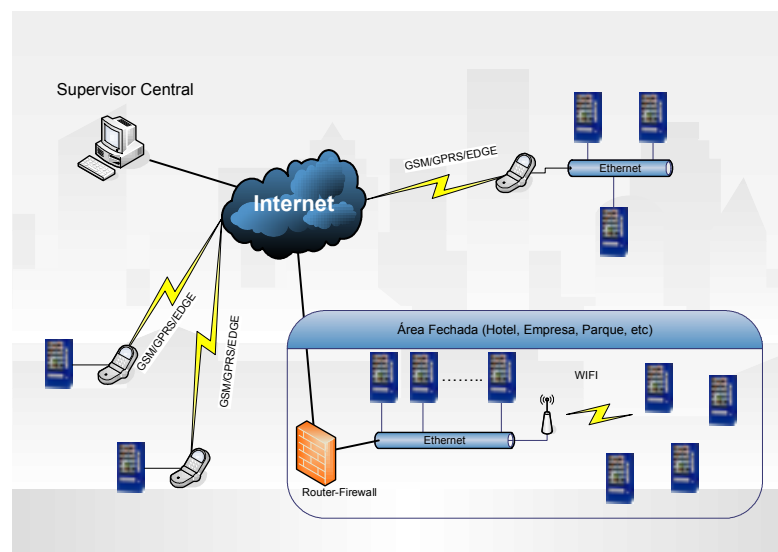
Figura 27 - Supervisão de máquinas de venda

O problema de controlar máquinas de venda (VM) espalhadas geograficamente afeta os supervisores das máquinas, os repositores de insumos, os técnicos de

manutenção, os proprietários dos estabelecimentos e os usuários das VM's. Seu impacto resulta em:

- Perda de vendas, decorrente da falta de produtos nas máquinas, ineficiência nos roteiros de visitação às máquinas para reposição e manutenção.
- Possibilidade de fraudes, roubos e vandalismos, decorrente da periodicidade inadequada das visitas para reposição e manutenção.
- Indisponibilidade de VM's para aquisição de produtos, devido à falha.
- Indisponibilidade de produtos para aquisição nas VM's, devido à falha de reposição de insumos.

Uma solução bem sucedida para o sistema de supervisão e controle de máquinas de venda proporcionaria os seguintes benefícios:

- Redução do índice de falta de produtos nas máquinas devido à reposição incorreta ou atrasada.
- Melhoria no planejamento dos roteiros de visita às máquinas para reposição e manutenção.
- Visualização centralizada e em tempo real do estoque e status das máquinas.
- Visualização centralizada de eventuais falhas e defeitos nas máquinas.
- Alta disponibilidade das VM's e dos produtos em estoque.

O sistema deve atender as seguintes necessidades dos envolvidos:

- Configurar VMs: permite especificar a configuração de cada VM a ser supervisionada (endereço de instalação, forma de conexão e tipo de produto).
- Status VM: permite obter os dados de cada VM em tempo real contendo informações de status de funcionamento e estoque de produtos.

- Relatório Período: permite obter o relatório de status de cada VM dentro de um período especificado.

4.2.3 Visão da Engenharia de Processos de Negócio (V2)

A atividade Modelagem Processos de Negócio (A2.1) associada à solução de controle de máquinas de venda teve como resultado a modelagem preliminar dos processos de negócio, contendo as principais funções com as suas entradas, saídas, controles e mecanismos, representada em IDEF 0 e apresentada na Figura 28.
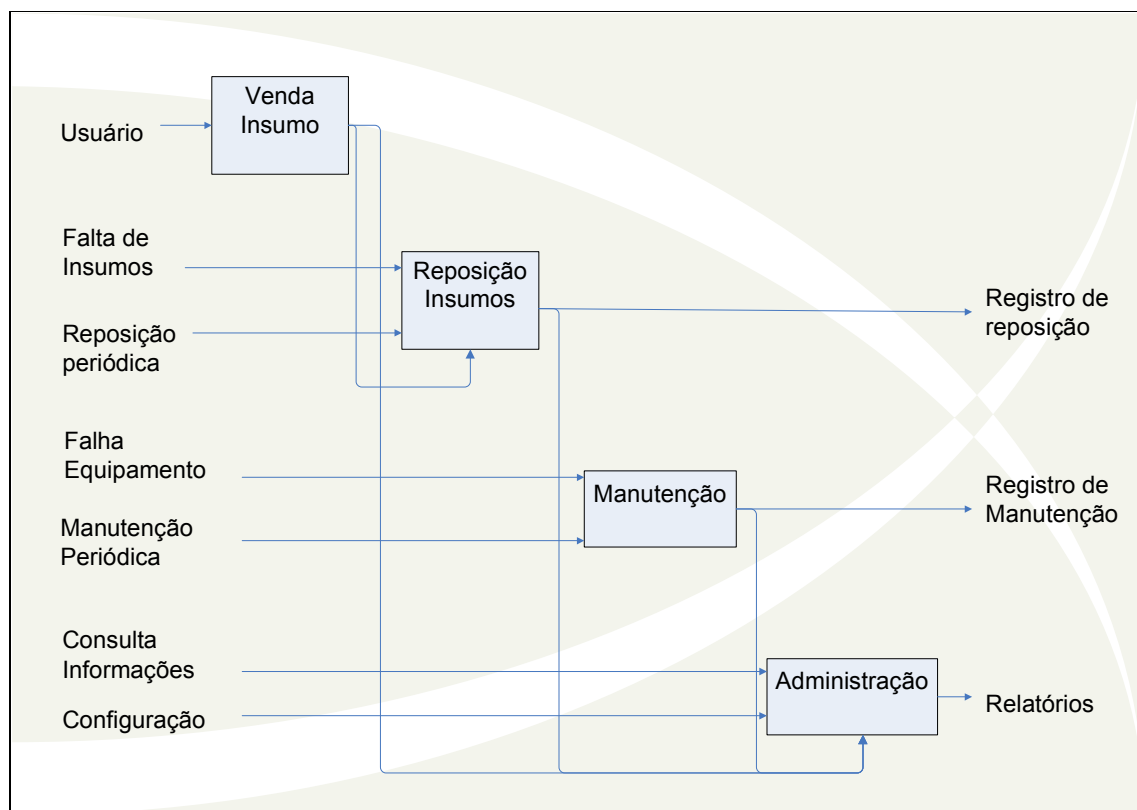
Figura 28 - VM: Representação dos processos de negócio em IDEF 0

A atividade Especificação Preliminar da Modelagem de Negócio em UML 2 (A2.2) neste caso não foi realizada.

A equipe de analistas de negócio formada por dois profissionais realizou a atividade de Modelagem Formal dos Requisitos de Negócio em SysML (A2.3), que tomou como base a representação em IDEF 0 e resultou no modelo de casos de usos de negócio representado em SysML. A Figura 29 apresenta o modelo de casos de usos de negócio obtido nesta etapa do trabalho.
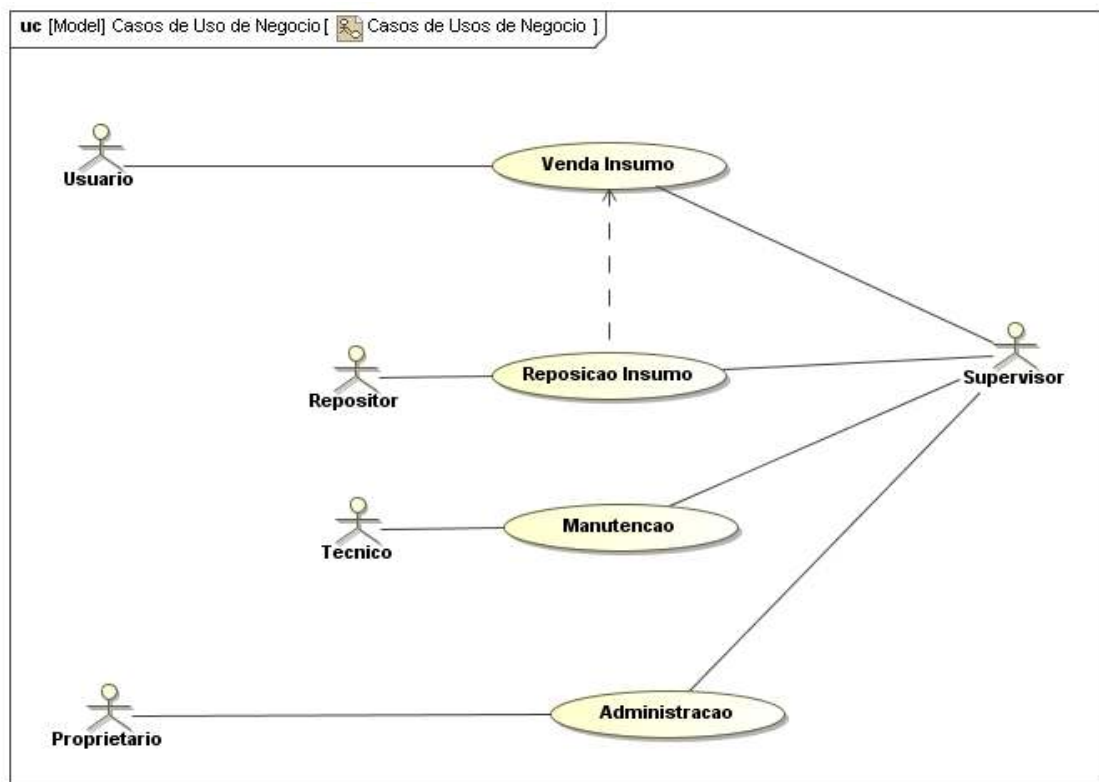
Figura 29 - VM: Casos de Usos de Negócio

Segue a descrição dos casos de uso de negócio associados à solução de controle de máquinas de venda:

- Venda Insumo: o usuário do sistema adquire insumos nas VM´s através da inserção de pagamento e retirada do produto desejado. O supervisor das VM´s obtém os resultados das vendas realizadas.
- Reposição Insumo: o responsável pela reposição dos insumos acessa as diversas VM´s repondo os mais variados tipos de produtos oferecidos

para venda. O supervisor obtém os resultados quanto a eventuais faltas de produtos e quanto ao histórico dos volumes de reposição para planejamento futuro das reposições.

- Manutenção: o técnico de manutenção executa os trabalhos de manutenção preventiva e corretiva das VM´s. O supervisor acompanha o status de funcionamento das VM´s e trata o planejamento das manutenções periódicas e roteiros de visitação dos técnicos.
- Administração: o proprietário do estabelecimento onde a VM está instalada e o supervisor podem obter relatórios de vendas, reposição, manutenção e falhas.

4.2.4 Visão da Engenharia de Requisitos (V1)

Para este estudo foi considerado apenas um dos produtos dentro da família de novos produtos em desenvolvimento, o Sistema de Supervisão de Máquinas de Venda de Café, onde a venda é realizada através do consumo de créditos armazenados em cartões indutivos.

As atividades Eliciação e Análise de Requisitos (A1.1) e Especificação Preliminar dos Requisitos em UML 2 (A1.2) do método de gestão de requisitos da Figura 12, neste caso, foram realizadas de forma parcial e diretamente em SysML. Uma descrição dos resultados intermediários, incluindo também atividades de análise e especificação do sistema, está apresentada no Apêndice A.

O resultado obtido na atividade Modelagem Formal dos Requisitos em SysML (A1.3), também da Figura 12, está representado pelo diagrama de requisitos e pelo diagrama de casos de uso do sistema.

A Figura 30 apresenta o diagrama de requisitos do Sistema de Supervisão de Máquinas de Venda de Café, obtido ao final da Modelagem Formal dos Requisitos (A1.3), onde os requisitos não funcionais estão modelados.
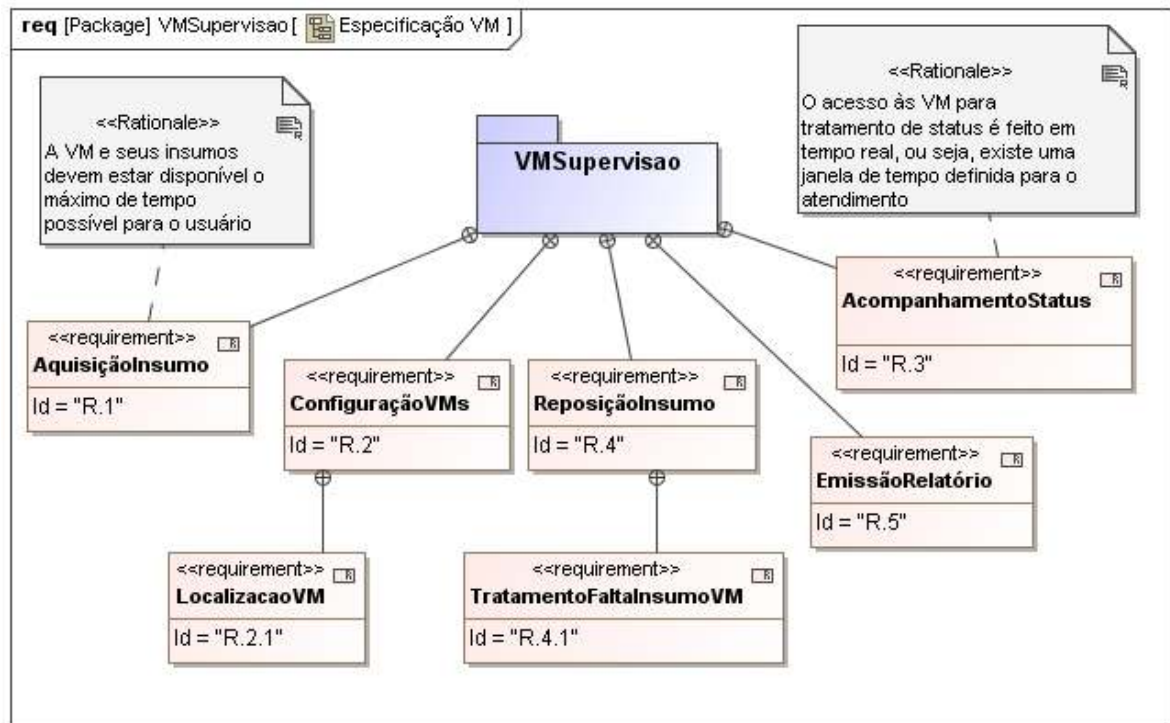
Figura 30 - Supervisão VM: Diagrama de Requisitos

A Figura 31 apresenta o diagrama de casos de uso associado ao Sistema de Supervisão de Máquinas de Venda de Café.
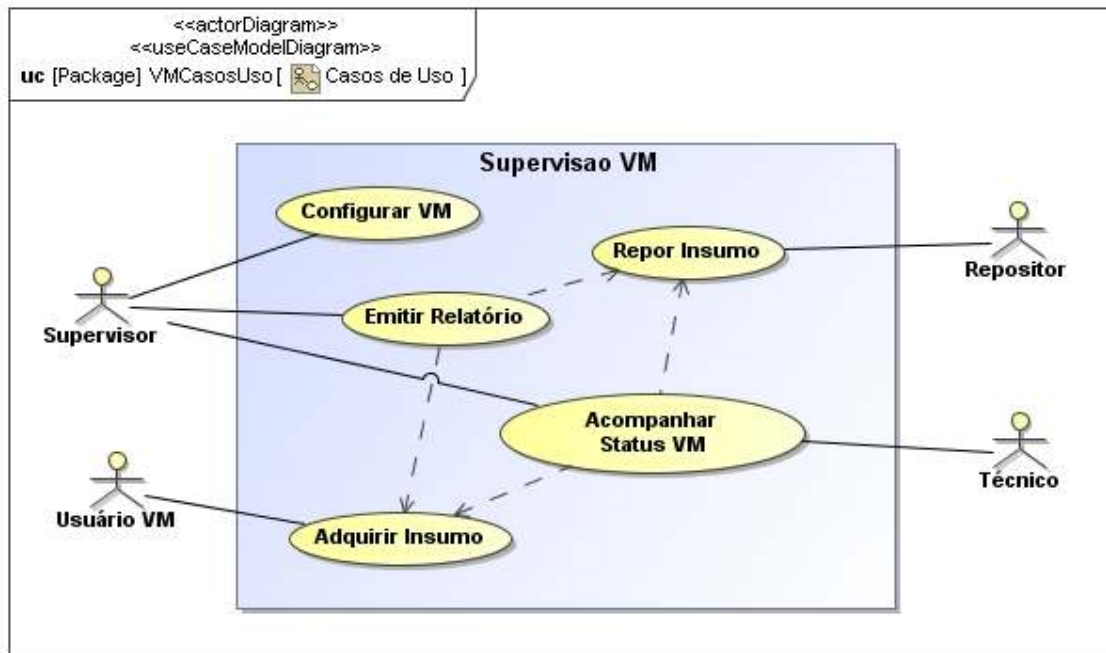
Figura 31 - Supervisão VM: Diagrama de Casos de Usos

Este sistema ainda está sendo especificado e as atividades associadas à Visão da Engenharia de Requisitos (V1) ainda não foram concluídas, pois o sistema está em fase de negociação de contrato junto ao cliente final do sistema, no caso a empresa que possui as máquinas de venda de café a serem automatizadas e supervisionadas.

4.2.5 Visão da Gestão de Requisitos do Sistema (V3)

Neste caso a atividade de Solicitação do Sistema (A3.1) da Figura 12 foi realizada após as atividades da Visão da Engenharia de Processos de Negócio (V2) e definiu o produto a ser desenvolvido visando a atender às necessidades específicas de um cliente já existente, onde as máquinas de venda de café já realizam a operação de venda através do consumo de créditos armazenados em cartões indutivos, mas não possuem supervisão automatizada. Foram definidas as ferramentas de controle de versão, adotados os modelos e padrões para os

documentos associados ao projeto, e no caso do SysML, foi utilizada a ferramenta MagicDraw para modelagem em SysML, apenas como experimentação para este estudo, pois a licença de utilização fornecida pelo fornecedor da ferramenta não permite a utilização para fins comerciais.

Na atividade Sincronização Gerencial de Conteúdo (A3.2) manteve as ferramentas de trabalho, o controle de versões, a padronização de documentos e o dicionário de dados e vocabulário utilizados na Visão da Engenharia de Requisitos (V1) e na Visão da Engenharia de Processos de Negócio (V2)

As atividades Consolidação dos Requisitos Alinhados com Negócio (A3.3), não possuem resultados a serem apresentados neste estudo e o mesmo ocorre com as atividades de Validação dos Requisitos do Sistema (A3.4) e Aceitação dos Requisitos do Sistema em SysML (A3.5), todas pertencentes à Visão da Gestão de Requisitos do Sistema (V3) do método proposto na Figura 12.

5 DISCUSSÃO

Considerando o objetivo deste trabalho que é propor um método de gestão que auxilie a comunicação, durante as etapas de definição dos requisitos de um sistema, entre a equipe de engenharia de requisitos e a equipe de engenharia de processos de negócio, através de uma maior aderência entre os requisitos do sistema e o modelo de negócio da organização, este capítulo apresenta e discute alguns tópicos associados ao método de gestão de requisitos proposto no capítulo 3, organizado pela discussão de cada uma das visões propostas no método, ou seja, visão da engenharia de requisitos, visão da engenharia de processos de negócio e visão da gestão dos requisitos do sistema.

5.1 VISÃO DA ENGENHARIA DE REQUISITOS

O referencial teórico apresentado no capítulo 2, Revisão da Literatura, sustenta as atividades propostas para esta visão através da abordagem do processo unificado de desenvolvimento de software (seção 2.3), engenharia de requisitos (seção 2.4) e linguagem SysML (seção 2.5).

A linguagem SysML foi escolhida como linguagem de modelagem formal devido à suas características gráficas, facilidade de aprendizado e por ser uma extensão da UML, herdando um potencial número de usuários. Este trabalho não tem como objetivo comparar e avaliar linguagens formais de representação.

Especificamente algumas atividades descritas no método proposto no capítulo 3 foram baseadas em itens específicos do referencial teórico. São elas:

- A atividade Eliciação e Análise dos Requisitos (A1.1), tratada na seção 3.1.1 deste estudo, está fundamentada e baseada no trabalho de Kotonya & Sommerville (1998).
- A atividade Especificação Preliminar dos Requisitos em UML (A1.2), tratada na seção 3.1.2 deste estudo, está baseada no processo unificado (Jacobson et al, 1999) (Kruchten, 2000) (Kroll, 2001) (Pressman,2006) (IBMRational, 2008) e na utilização da linguagem UML (Booch et al, 1999) (Gilleanes, 2004).
- A atividade Modelagem Formal dos Requisitos em SysML (A1.3), tratada na seção 3.1.3 deste estudo, está baseada na característica da linguagem SysML de ser uma extensão da UML 2 (OMGSysML, 2006) (Balmelli, 2006), nos recursos de interoperabilidade (OMGMOF, 2007) e nas características formais de SysML apresentadas na seção 2.5.4 e pela RFP que originou a linguagem SysML (OMGRFP, 2003).

A realização das atividades propostas para a Visão da Engenharia de Requisitos associado com a atuação gerencial desempenhada pela Visão da Gestão dos Requisitos, propicia a antecipação da documentação e formalização dos requisitos do sistema, pois garante uma maior aderência e sincronização com os processos de negócio e uma conseqüente sustentação para a negociação junto aos stakeholders dos requisitos sistema.

5.2 VISÃO DA ENGENHARIA DE PROCESSO DE NEGÓCIO

As atividades associadas à Visão da Engenharia de Processo de Negócio, integrante do método para gestão de requisitos descrito no capítulo 3, foram obtidas a partir da pesquisa bibliográfica retratada no capítulo 2, Revisão da Literatura,

especificamente na seção 2.2, Engenharia de Processos de Negócio, para as atividades Modelagem de Processos de Negócio (seção 3.2.1) e Especificação Preliminar da Modelagem de Negócio em UML (seção 3.2.2), em que o estudo realizado por Vicente (2004) foi utilizado como referência para o desenvolvimento destas duas atividades, fornecendo um método para modelagem de processos e requisitos de negócio que permite apoiar a especificação dos requisitos de sistema utilizando UML.

A atividade Modelagem Formal dos Requisitos de Negócio em SysML (A2.3), apresentada na seção 3.2.3, está baseada na característica da linguagem SysML de ser uma extensão da UML 2 (OMGSysML, 2006) (Balmelli, 2006), nos recursos de interoperabilidade (OMGMOF, 2007) e nas características formais de SysML apresentadas na seção 2.5.4 e pela RFP que originou a linguagem SysML (OMGRFP, 2003).

Existe a possibilidade das atividades A2.1 e A2.2 serem realizadas antecipadamente ao tratamento dos requisitos do sistema, podendo ocorrer até mesmo que a modelagem de negócio não seja realizada. Neste caso, o método de gestão de requisitos descrito no capítulo 3, prevê que uma equipe de trabalho seja alocada para a realização das atividades A2.1 e A2.2, ocasionando um possível aumento dos custos das fases iniciais do projeto.

5.3 GESTÃO DOS REQUISITOS DO SISTEMA

As atividades associadas à Visão da Gestão dos Requisitos do Sistema, integrante do método para gestão de requisitos descrito no capítulo 3, foram obtidas a partir da pesquisa bibliográfica retratada no capítulo 2, Revisão da Literatura, especificamente na seção 2.3, Processo unificado, e na seção 2.4.5, Gerenciamento

de requisitos. A experiência prática do autor deste estudo no desenvolvimento de sistemas de informação e de automação, atuando desde a implementação, especificação e gestão de projetos, também foi utilizada para incorporar características operacionais e necessidades que são encontradas em projetos reais.

Leffingwell & Widrig (2000) propõem um processo para gerenciamento da engenharia de requisitos baseado na formação adequada das equipes de desenvolvimento, processo adotado neste trabalho através da separação explícita, mas gerenciada, entre os engenheiros de requisitos e os analistas de negócio, atuando respectivamente na Visão da Engenharia de Requisitos e na Visão da Engenharia de Processos de Negócio, permitindo que as duas equipes trabalhem de forma independente, mas sincronizadas pela equipe de gestão.

A atividade de consolidação dos requisitos do sistema alinhados com os processos de negócio, descrita na seção 3.3.3, utiliza os recursos oferecidos pelas ferramentas CASE de modelagem SysML para verificação de diferenças sintáticas entre dois modelos para tratar as eventuais inconsistências existentes entre o modelo em SysML obtido como resultado da Visão da Engenharia de Requisitos e o modelo obtido na Visão da Engenharia de Processos de Negócio, para que de forma iterativa as inconsistência possam ser tratadas e um modelo de requisitos do sistema em SysML, aderente aos processos de negócio possa ser obtido através do recurso de fusão de modelos, também oferecido pelas ferramentas CASE de modelagem SysML.

Cabe ressaltar que a atividade de consolidação dos requisitos do sistema, alinhada com os processos de negócio, necessita ser aplicada a casos reais, pois este estudo apresenta apenas uma experiência, descrita no capítulo 4, cujo objetivo foi explicitar o método proposto e não demonstrar os artefatos obtidos.

As atividades de validação e aceitação dos requisitos do sistema em SysML, que estão sendo descritas, respectivamente, nas seções 3.3.4 e 3.3.5, estão fundamentadas e baseadas no trabalho de Kotonya & Sommerville (1998).

Estima-se um maior retorno de investimento, com a utilização do método de gestão proposto no capítulo 3, quanto maior e mais complexo for o projeto, devido à dificuldade de verificação manual da aderência entre os requisitos do sistema e os processos de negócio.

A antecipação da documentação e formalização dos requisitos, a aderência aos processos de negócio, a negociação e aprovação dos resultados parciais junto aos stakeholders, a representação dos requisitos em uma linguagem formal reduzindo eventuais ambigüidades, favorecem que as fases iniciais do projeto ganhem uma maior importância estratégica dentro do ciclo de vida do projeto e, possivelmente, um incremento nos custos para a realização destas fases. Entretanto oferecem um menor risco de ocorrência de problemas nas fases seguintes à definição dos requisitos com uma conseqüente redução dos custo das fases de design, implementação, teste, implantação, produção e manutenção do sistema, lembrando que de acordo com Carr (2000) a correção de problemas de requisitos realizados na fase de produção do sistema produção pode custar 500 vezes mais do que se fossem corrigidos na fase de requisitos.

6 CONCLUSÕES E TRABALHOS FUTUROS

O objetivo deste trabalho, propor um método de gestão que auxilie a comunicação, durante as etapas de definição dos requisitos de um sistema, entre a equipe de engenharia de requisitos e a equipe de engenharia de processos de negócio, através de uma maior aderência entre os requisitos do sistema e o modelo de negócio da organização, foi alcançado através da revisão literária realizada, do método de gestão de requisitos proposto no capitulo 3, da adoção de SysML como linguagem formal de modelagem.

A partir do método de gestão de requisitos proposto no capitulo 3 e dos casos apresentados como ilustração e descritos no capítulo 4, conclui-se que a comunicação entre as equipes será mais efetiva e produtiva, pois o método procura evoluir gradativamente, e de forma gerenciada, o grau de aderência, quanto aos requisitos do sistema, entre a visão da engenharia de requisitos e a visão da engenharia de processos de negócio, chegando, nas últimas etapas do método, a uma representação única dos requisitos na linguagem formal SysML.

A comunicação com os *stakeholders* será mais efetiva e produtiva, podendo resultar em um processo de aceitação e validação dos requisitos mais eficiente, convergente e, possivelmente, com tempos de aceitação final mais curtos, pois o método proposto oferece uma maior aderência dos requisitos do sistema com os processos de negócio da organização, estabelece a realização de validações intermediárias dos resultados parciais obtidos e utiliza linguagem formal para a modelagem dos requisitos permitindo grande redução das ambigüidades no entendimento e interpretação semântica do modelo de requisitos.

A utilização da linguagem formal SysML para modelagem dos requisitos poderá também otimizar o trabalho a ser realizado pelas equipes envolvidas com as demais etapas do ciclo de vida do desenvolvimento do sistema, como *design*, implementação, teste e implantação, pois oferece como insumo para estas etapas uma modelagem dos requisitos com uma possível redução no grau de ambigüidades e necessidade de intervenção tanto dos engenheiros de requisitos como dos analistas de negócio.

A principal limitação deste trabalho é o fato de o método proposto no capítulo 3 não ter sido aplicado a um projeto em sua totalidade. A aplicação do método proposto a casos passados limita a verificação do impacto sobre as equipes de trabalho e sobre o andamento do projeto. Uma nova pesquisa de campo mais profunda e detalhada, envolvendo diversos estudos de caso, associados à aplicação real e completa do método proposto no capítulo 3, poderá oferecer diversos resultados, como:

- A verificação e medição das melhorias obtidas quanto à comunicação entre a equipe de requisitos e a equipe de modelagem de processos, durante as atividades de especificação dos requisitos do sistema.
- A verificação e medição das melhorias obtidas quanto ao tempo de validação e aceitação dos requisitos do sistema pelos *stakeholders*.
- Evolução e comprovação das atividades do método proposto no capítulo 3, associadas à visão da gestão dos requisitos, principalmente quanto à consolidação dos requisitos do sistema alinhados com os processos de negócio.

As conclusões obtidas e a limitação indicada para este trabalho sugerem a realização de novos trabalhos e estudos, tais como:

- Desenvolver técnicas de consolidação e sincronização dos requisitos do sistema alinhados aos processos de negócio, utilizando ferramentas automatizadas baseadas em linguagens formais de modelagem.
- Desenvolver pesquisa visando à verificação da efetividade do método proposto em casos reais, através da medição e verificação dos resultados obtidos.
- Desenvolver pesquisa visando estender a utilização de linguagem formal para a etapa de análise dos requisitos.

REFERÊNCIAS

Akhavan, Peyman; Jafari, Mostafa; Ali-Ahmadi, Ali R. "Exploring the interdependency between reengineering and information technology by developing a conceptual model", Business Process Management Journal, Vol. 12 No. 4, 2006

Alencar, Fernanda Maria Ribeiro de "Mapeando a modelagem organizacional em especificações precisas", Tese (Doutorado) - Universidade Federal de Pernambuco, 1999

Alencar, Fernanda; Castro, Jaelson; Cysneiros, Gilberto; Mylopoulos, John "From Early Requirements Modeled by the i* Technique to Later Requirements Modeled in Precise UML", WER2000, 2000

Andersson, Birger; Bider, Ilia; Johannesson, Paul; Perjons, Erik "Towards a formal definition of goal-oriented business process patterns", Business Process Management Journal, Vol. 11 No. 6, 2005

ARTISAN “ARTiSAN Software Tools”, site da organização, 2008. Disponível em: http://www.artisansw.com/ (Acesso 28/07/2008).

Baker, Bryon "Business Modeling with UML: The Light at the End of the Tunnel", Rational University, 2001.

Balmelli, Laurent "An Overview of the Systems Modeling Language for Products and Systems Development", IBM Technical Report, 2006.

Barros, Oscar "Business process patterns and frameworks", Business Process Management Journal, Vol. 13 No. 1, 2007

Boehm, B.W., “A Spiral Model of Software Development and Enhancement”, IEEE Computer, vol 21, nº 5, p. 61-72, 1988.

Booch, Grady; Rumbaugh, James; Jacobson, Ivar "The Unified Modeling Language User Guide", Addison-Wesley, 1999.

Bosik, D.; Terdiman, R. - “Outsourcing Application: The Time Has Come”, Strategic Analysis Report, Gartner Group, Julho 1997

Cameira, R.; Caulliraux H. “Engenharia de processos de negócios: considerações metodológicas com vistas à análise e integração de processos”, Simpósio de Administração da Produção, Logística e Operações Internacionais, São Paulo, 2000.

Carr, Joseph J. "Requirements engineering and management: the key to designing quality complex systems", The TQM Magazine, Volume 12, Number 6, 2000

Corder, S. "Indústria têxtil: inovações tecnológicas e impactos sobre as qualificações dos trabalhadores", Dissertação (Mestrado) – Instituto de Geociências, Universidade Estadual de Campinas, 1994.

Cundiff, R.; Lusher, C.; Martinez, D.; Pucciarelli, J.; Purchase, E. - “RFPs for Customer Service and Support Applications: A Cookbook and an Example”, Strategic Analysis Report, Gartner Group, Abril 1997

Damij, Nadja "Business process modelling using diagrammatic and tabular techniques", Business Process Management Journal, Vol. 13 No. 1, 2007

Davenport, Thomas H. "Reengenharia de Processos", Editora Campus, 1994.

De Sordi, José O.; Spelta, Andrea G. "Análise de componentes da tecnologia de Business Process Management Systems (BPMS) sob a perspectiva de um caso prático", Revista de Gestão da Tecnologia e Sistemas de Informação, Journal of Information Systems and Technology Management, Vol. 4, No. 1, p. 71-94, 2007.

Dias, Felipe; Morgado, Gisele; Oscar, Pedro; Silveira, Denis; Alencar, Antonio Juarez; Lima, Priscila; Schmitz, Eber "Uma Abordagem para a Transformação Automática do Modelo de Negócio em Modelo de Requisitos", WER06 - Workshop em Engenharia de Requisitos, 2006

Dym, Clive L.; Little, Patrick "Engineering Design: A Project-Based Introduction", John Wiley & Sons, 2000

EmbeddedPlus “EmbeddedPlus Engineering”, site da organização, 2008. Disponível em: http://www.embeddedplus.com/ (Acesso 28/07/2008).

Eriksson, H.; Penker, M. “Business Modeling with UML”, 1999. Disponível em: http://www.therationaledge.com. (Acesso 23/09/2003).

Frappier, M.; Habrias, H. “Software Specification Methods: An Overview Using a Case Study”, Springer, 2000.

Gilleanes, G. "UML uma abordagem prática", Novatec Editora, 2004.

Harmon, Paul "The OMG's Model Driven Architecture and BPM", Business Process Trends, volume 2, número 5, 2004.

IBMRational “IBM Rational Software”, site da organização, 2008. Disponível em: http://www-306.ibm.com/software/br/rational/ (Acesso 28/07/2008).

IDEF "Integrated Definition Methods", site da organização, 2007. Disponível em http://www.idef.com/ (Acesso 14/05/2007).

IDS “ARIS Solutions - IDS Schees AG”, site da organização, 2007. Disponível em http://www.ids-scheer.com/ (Acesso 14/05/2007)

INCOSE “The International Council on Systems Engineering”, site da organização, 2007. Disponível em: http://www.incose.org/ (Acesso 14/05/2007).

Jacobson, I.; Booch, G.; Rumbaugh, J “The unified software development process”, Addison-Wesley, 1999

Jackowski, Z. "Business Modeling with UML: A Business Process Centred Architecture", (Disponível em: http://www.agilealliance.org/show/1202, Acesso em: 04/07/2007), 2003

Kerzner, H. "Project Management: A Systems Approach to Planning, Scheduling, and Controlling", 7a edição, John Wiley & Sons, 2001.

Kintschner, F. "Método de Modelagem de Processos para Apoio ao Desenvolvimento de Software", Tese (Doutorado) - Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2003

Korthaus, A. "Using UML for Business Object Based Systems Modeling", The Unified Modeling Language --- Technical Aspects and Applications, pages 220--237. Physica-Verlag, Heidelberg, Germany, 1998.

Kotonya, G.; Sommerville, I. "Requirements Engineering: Processes and Techniques", John Wiley & Sons Ltd., 1998.

Kroll, P., “The Spirit of RUP”, The Rational Edge, IBM, 2001.

Kruchten, P. "The Rational Unified Process: An Introduction", segunda edição, Addison-Wesley, 2000

Leffingwell, D.; Widrig, D. "Managing Software Requirements: a unified approach", Addison-Wesley, 2000

Maiden, N.; Rugg, G."ACRE: selecting methods for requirements acquisition", Software Engineering Journal, v.11 n. 3, pg 183-192, 1996.

Maiden, N.; Rugg, G.; Patel, P. "Guidelines for better scenarios: supporting theories and evidence", Tenth International Workshop on Database and Expert Systems Applications, 1999.

Marshall, C. "Enterprise Modeling with UML", Addison-Wesley, 2000

Molokken, K.; Jorgensen, M. "A Review of Surveys on Software Effort Estimation", Proceedings of the 2003 International Symposium on Empirical Software Engineering (ISESE’03), 2003.

NoMagic “No Magic”, site da organização, 2008. Disponível em: http://www.nomagic.com/ (Acesso 28/07/2008).

Noran, O. "Business Modeling: UML vs. IDEF", School of Computing and Information Technology, Griffith University, 2000

Nuseibeh, B.; Easterbrook, S. "Requirements engineering: a roadmap", International Conference on Software Engineering, Proceedings of the Conference on The Future of Software Engineering, pg 35 - 46, 2000.

OMG “Object Management Group”, site da organização, 2007. Disponível em: http://www.omg.org/ (Acesso 14/05/2007).

OMGMDA “Model Driven Architecture - MDA”, Object Management Group, site da organização, 2007. Disponível em: http://www.omg.org/mda/ (Acesso 26/05/2007).

OMGMOF “Meta-Object Family”, Object Management Group, site da organização, 2007. Disponível em: http://www.omg.org/mof/ (Acesso 26/05/2007).

OMGRFP “UML for Systems Engineering RFP”, Object Management Group, 2003. Disponível em: http://www.omg.org/ (Acesso 14/maio/2007).

OMGSysML “OMG Systems Modeling Language (OMG SysML) Specification”, ptc/06-05-04 Final Adopted Specification, Object Management Group, 2006. Disponível em: http://www.omgsysml.org/ (Acesso 14/maio/2007).

OMGUML “Unified Modeling Language”, site da organização, 2007. Disponível em: http://www.uml.org/ (Acesso 14/05/2007).

Papyrus “Papyrus UML”, site da organização, 2008. Disponível em: http://www.papyrusuml.org/ (Acesso 28/07/2008).

Peak, R.; Burkhart, R.; Friedenthal, S.; Wilson, M.; Bajaj, M.; Kim, I. "Simulation-Based Design Using SysML Part 1: A Parametrics Primer", INCOSE Intl. Symposium, San Diego, 2007a.

Peak, R.; Burkhart, R.; Friedenthal, S.; Wilson, M.; Bajaj, M.; Kim, I. "Simulation-Based Design Using SysML Part 2: Celebrating Diversity by Example", INCOSE Intl. Symposium, San Diego, 2007b.

Pressman, R. "Engenharia de Software", sexta edição, McGraw-Hill, 2006

Procaccino, J.; Verner, J.; Overmyer, S.; Darter, M. ”Case study: factors for early prediction of software development success”, Information and Software Technology, v.44, p.53-63, 2002.

Rukanova, B. "BUSINESS TRANSACTIONS AND STANDARDS", Tese (Doutorado)-University of Twente, 2005

Rukanova, B.; Slooten, K.; Stegwee, R. "Business Process Requirements, Modeling Technique, and Standard: How to identify interoperability gaps on a process level", Interopesa, 2005.

Russell, N.; van der Aalst, W.; ter Hofstede, A.; Wohed, P. "On the Suitability of UML 2.0 Activity Diagrams for Business Process Modelling", In Third Asia-Pacific Conference on Conceptual Modelling (APCCM2006), volume 53 of CRPIT, pages 95–104, 2006.

Salm, J. "Extensões da UML para descrever processos de negócio", Dissertação (Mestrado) - Engenharia de Produção), Universidade Federal de Santa Catarina, Florianópolis, 2003.

Santos, R.; Cameira, R.; Clemente, A.; Clementa, R. "Engenharia de processos de negócios: aplicações e metodologias", ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, Curitiba, 2002.

Selic, B. "Tutorial: an overview of UML 2", 28th International Conference on Software Engineering, 2006.

Shaw, D.; Holland, C.; Kawalek, P.; Snowdon, B; Warboys, B. "Elements of a business process management system: theory and practice", Business Process Management Journal, Vol. 13 No. 1, 2007

Shaw, M.; Gaines, B. "Requirements Acquisition", Software Engineering Journal, V.11 N.3, pg 149-165, 1996.

Standish Group “Charting the Seas of Information Technology – Chaos”, The Standish Group International, 1995.

SysMLForum “SysML Forum”, site da organização, 2007. Disponível em: http://www.sysmlforum.com/ (Acesso 14/05/2007).

Van Lamsweerde, A.; Darimont, R.; Letier, E. "Managing conflicts in goal-driven requirements engineering", IEEE Transactions on Software Engineering, v24, n11, pg 908-926, 1998.

Vanderperren, Y.; Dehaene, W. "SysML and Systems Engineering Applied to UML-Based SoC Design", Design Automation Conference (DAC), UML for SoC Design Workshop, 2005.

Vicente, L. "Modelagem de Processos e Linguagem de Modelagem Unificada (UML) uma análise crítica", Dissertação (Mestrado) - Engenharia de Produção, Universidade Federal do Rio de Janeiro, 2004.

Vernadat, F. "Enterprise modeling and integration: principles and applications", London, Chapman & Hall, 1996.

Viller, S; Sommerville, I. "Social Analysis in the Requirements Engineering Process: from ethnography to method", 4th International Symposium on Requirements Engineering (RE'99), 1999.

W3CXML “World Wide Web Consortium, XML”, site da organização, 2008. Disponível em: http://www.w3.org/XML/ (Acesso 05/03/2008).

APÊNDICE A – Supervisão VM: Especificação em SysML

Este apêndice apresenta a especificação preliminar em SysML do sistema de controle e supervisão de máquinas de vendas utilizado como elemento de apoio para o processo de eliciação dos requisitos do sistema.

Na especificação de um sistema a primeira tarefa é decidir o que pertence ao sistema e o que não pertence. Para este fim, utiliza-se o diagrama informal de contexto do sistema, apresentado na Figura 32, onde o domínio e as fronteiras do sistema são representados. Neste caso, as operadoras de serviço celular, provedoras de acesso à Internet e os equipamentos de máquina de venda propriamente ditos, são considerados externos, pois o sistema trata a supervisão dos equipamentos e não o seu funcionamento e operação.
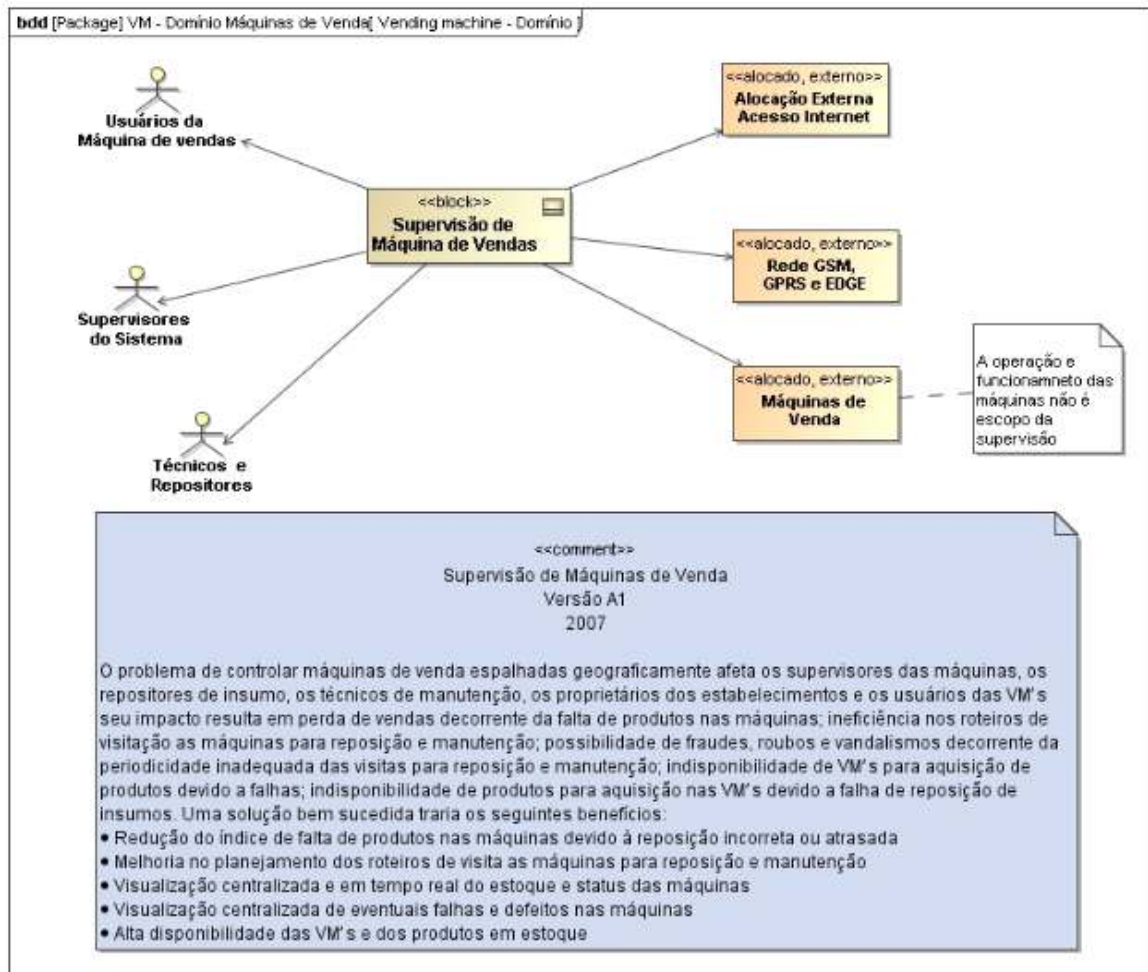
Figura 32 - Domínio e as fronteiras do sistema

A Figura 33 apresenta o diagrama de requisitos do sistema de supervisão de máquinas de venda, indicando dependências entre requisitos e seus racionais.
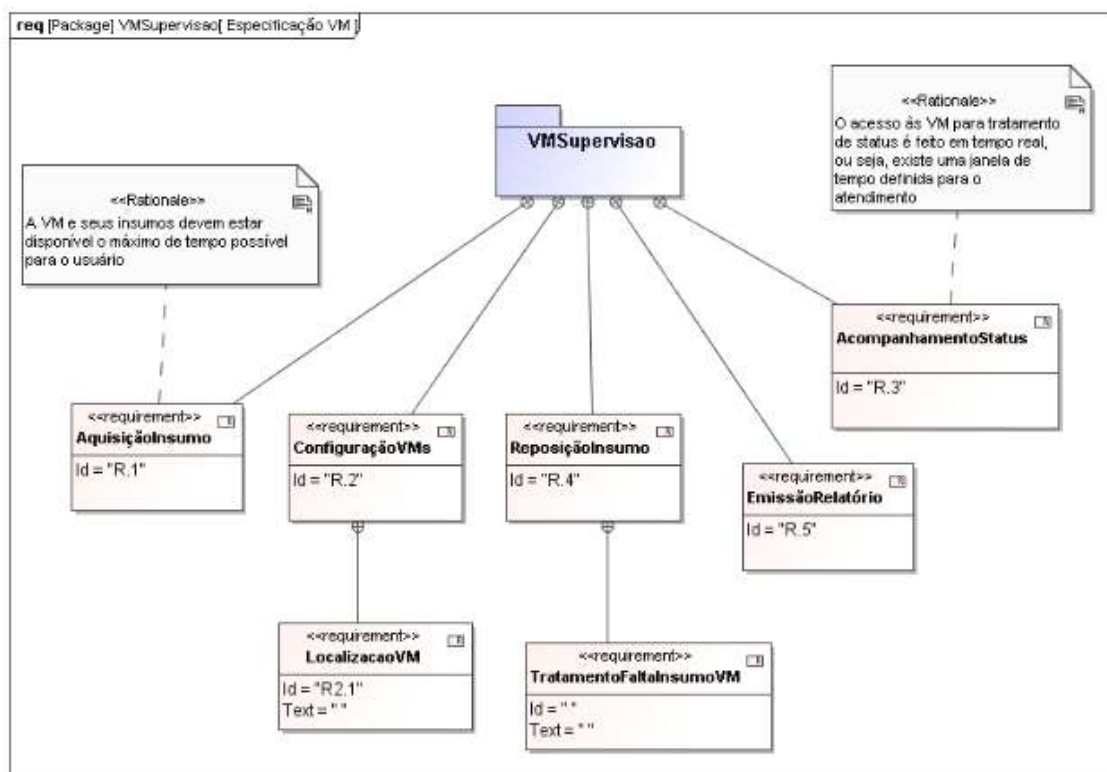
Figura 33 - Diagrama de Requisitos

A Figura 34 apresenta o diagrama de casos de uso contendo as relações e dependências entre os casos de uso e os atores do sistema.
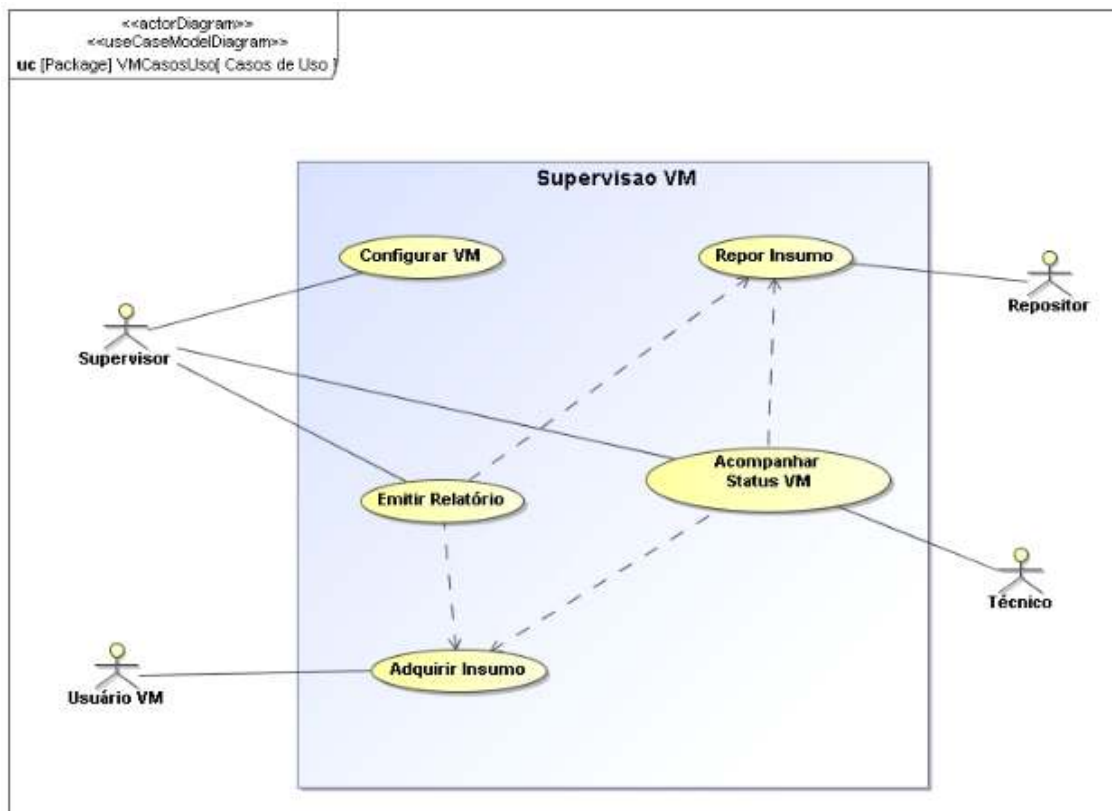
Figura 34 - Diagrama de Casos de Uso

A Figura 35 apresenta um diagrama de seqüência em linguagem SysML, onde podemos observar a estrutura e a representação das ações.
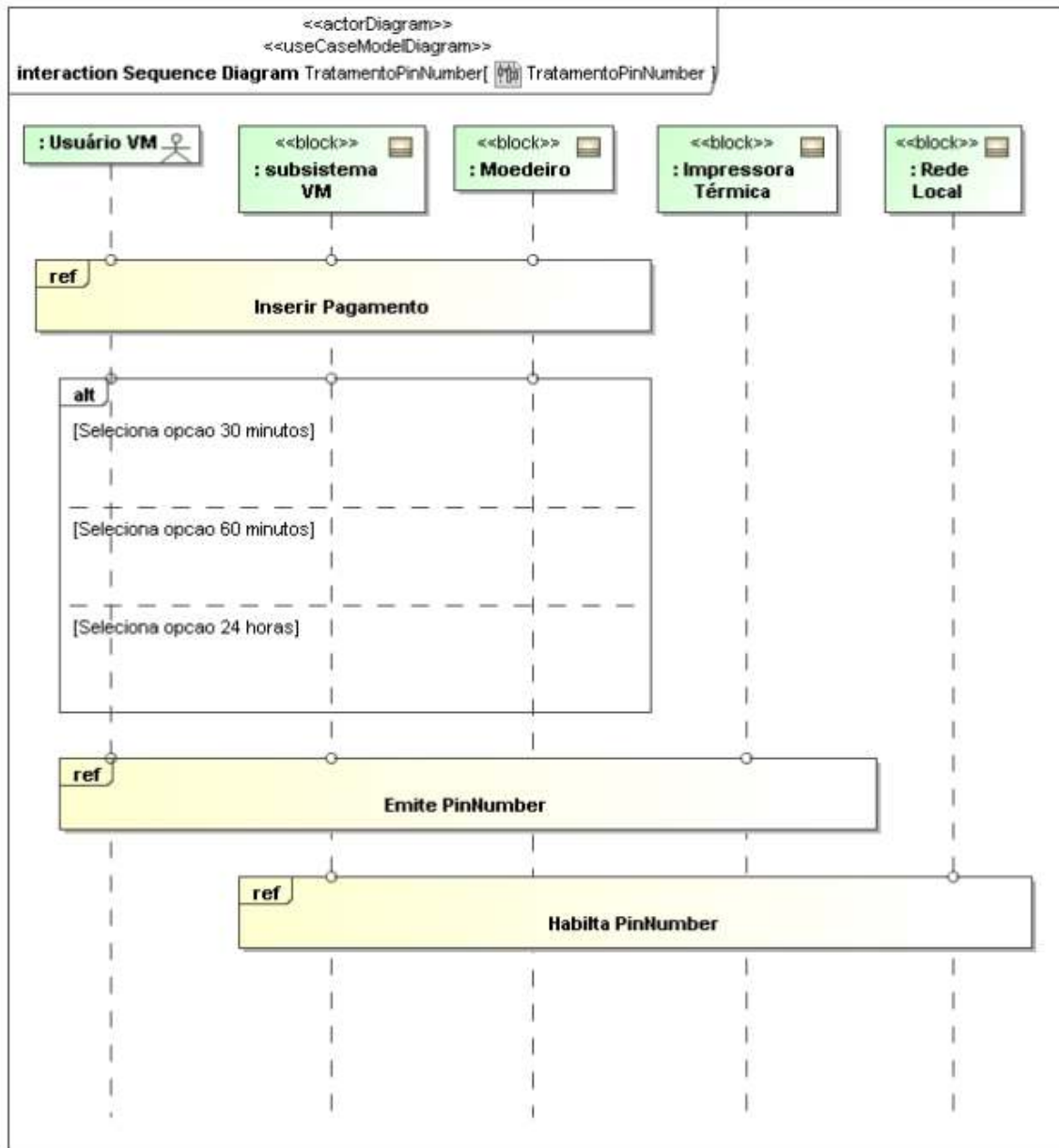
Figura 35 - Diagrama de Seqüência em SysML

O diagrama de máquina de estado da Figura 36 representa os estados e transições do caso de uso Acompanhamento Status.
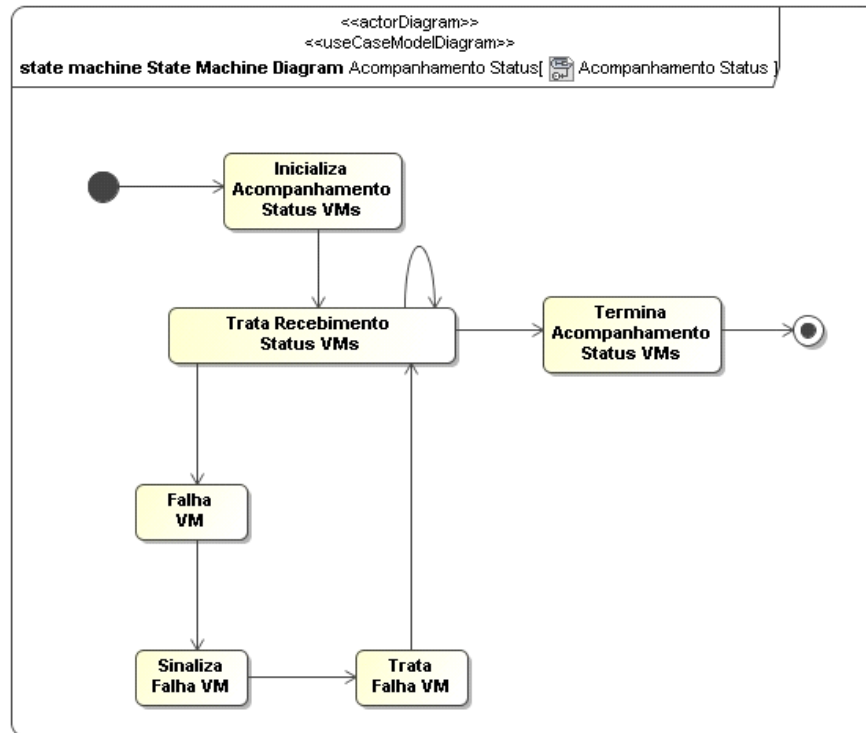
Figura 36 - Diagrama de Máquina de Estado

O diagrama de parâmetro (Figura 37) representa as restrições impostas ao tratamento do pagamento, podendo definir desempenho, confiabilidade e propriedades físicas.
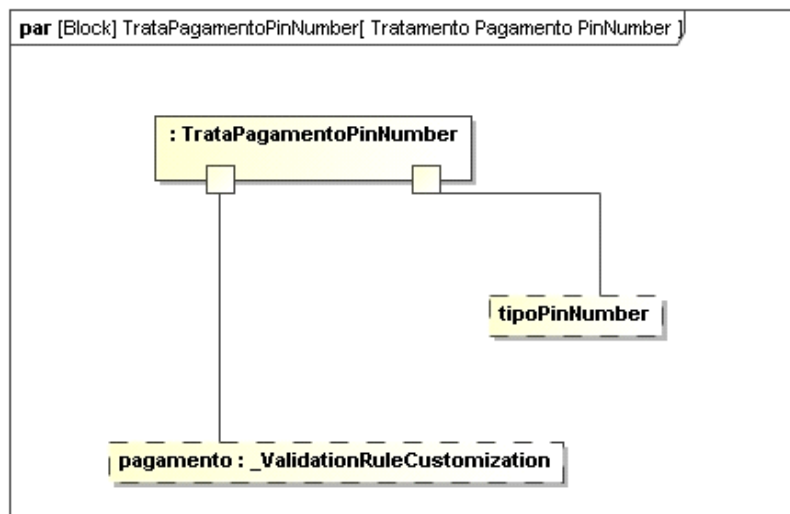
Figura 37 - Diagrama de Parâmetro

O diagrama de pacotes do modelo (Figura 38) mostra a estrutura utilizada para avaliar o problema, com os relacionamentos entre os pacotes e as visões.
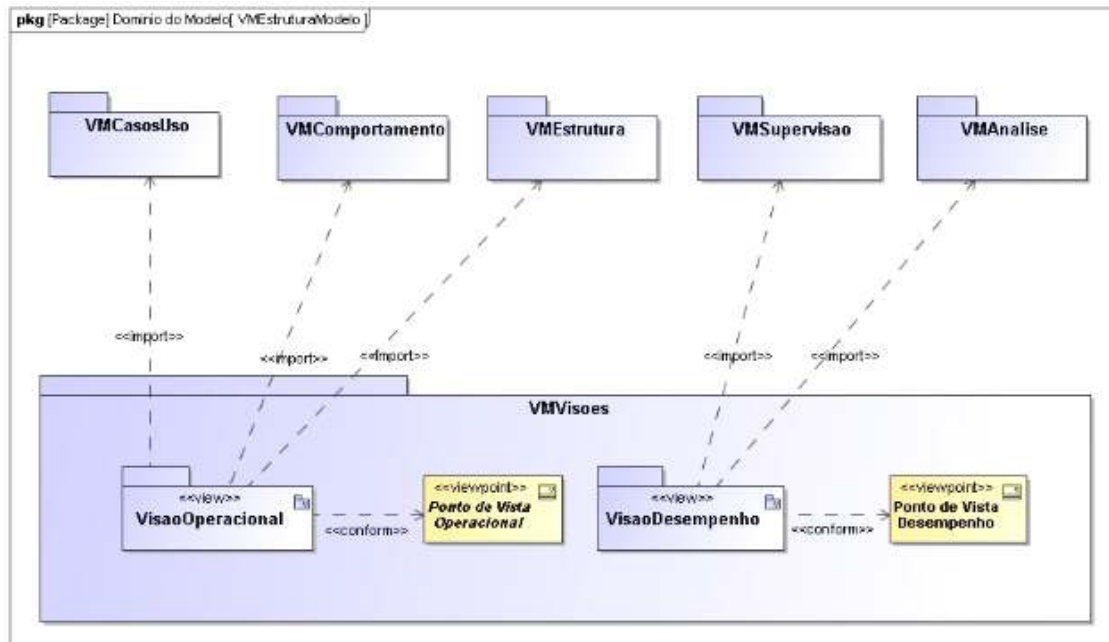
Figura 38 - Estrutura de Pacotes do Modelo

O diagrama de domínio hierárquico define formalmente o contexto e as fronteiras do sistema (Figura 39).
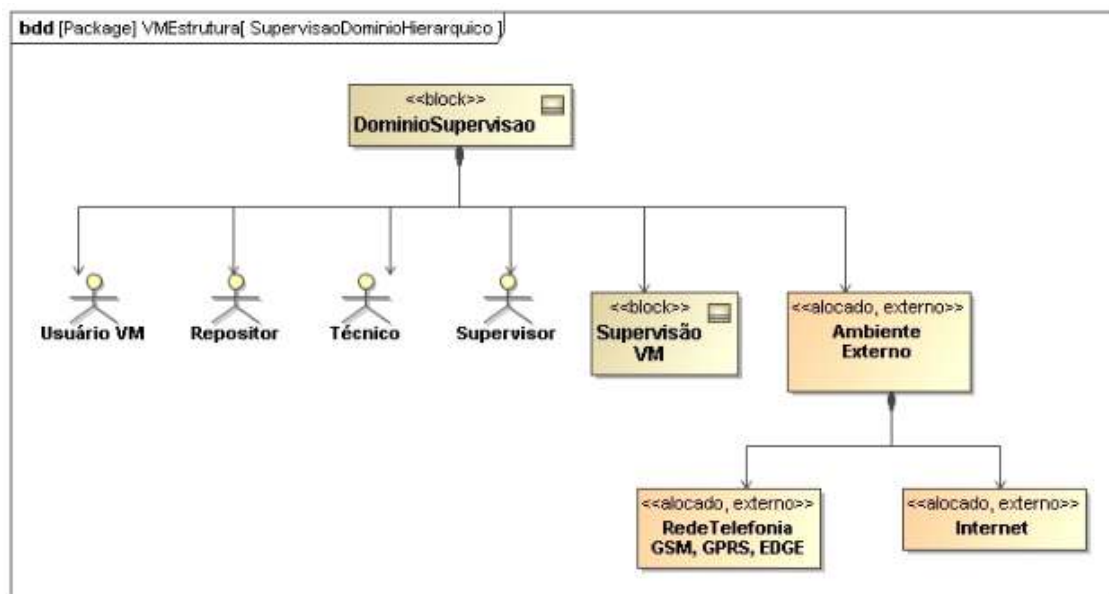
Figura 39 - Domínio Hierárquico

A estrutura hierárquica do sistema define os blocos que compõem o sistema e os sub-sistemas (Figura 40).
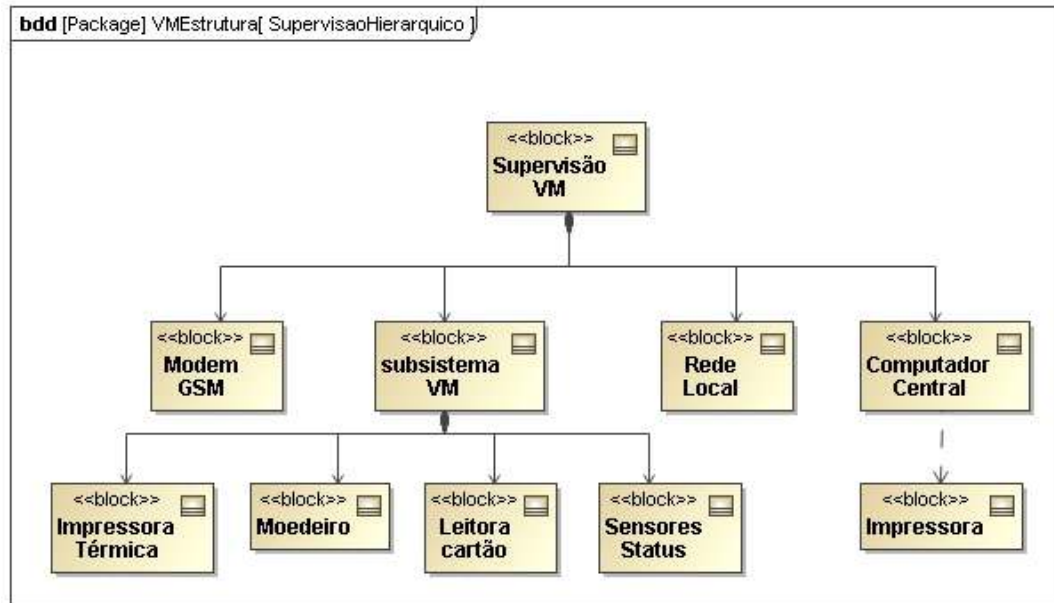
Figura 40 - Estrutura Hierárquica do Sistema

O diagrama de pacotes é utilizado na Figura 41 para definir a visão de desempenho associada à aquisição de insumos pelo usuário do sistema nas máquinas de venda.
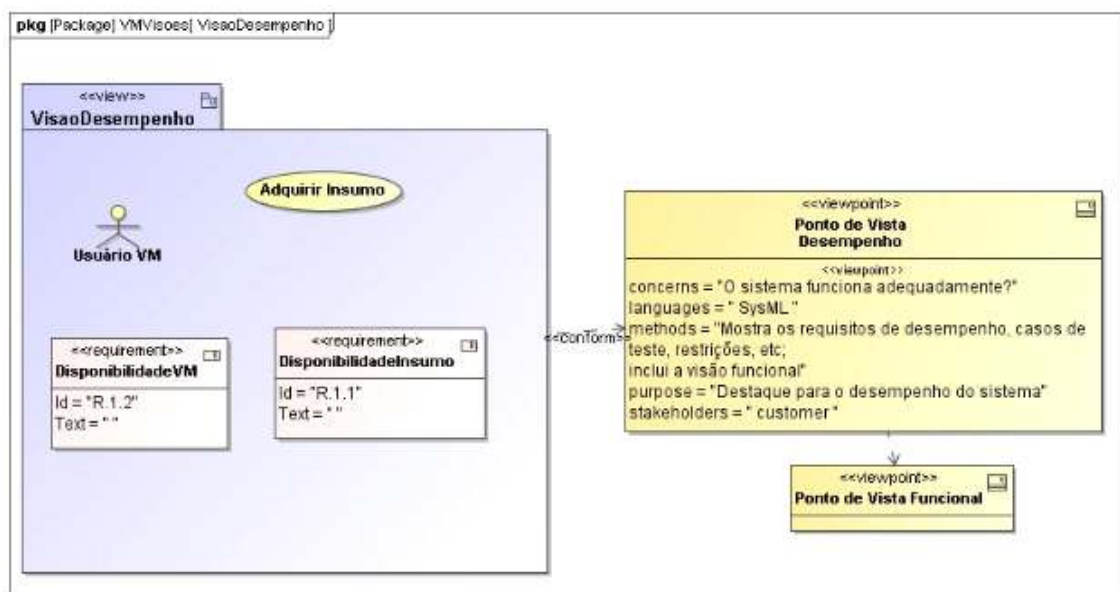
Figura 41 - Visão de desempenho